

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
ІМ. ІГОРЯ СІКОРСЬКОГО»  
Факультет інформатики та обчислювальної техніки  
Кафедра обчислювальної техніки**

До захисту допущено:

Завідувач кафедри

Сергій СТИРЕНКО

«\_\_» \_\_\_\_\_ 20\_\_ р.

**Дипломний проєкт  
на здобуття ступеня бакалавра  
за освітньо-професійною програмою «Комп'ютерні системи та мережі»  
спеціальності 123 «Комп'ютерна інженерія»  
на тему: «Роботизована рука»**

Виконала:

студентка IV курсу, групи ІО-64

Анастасія НЕВМЕРЖИЦЬКА

\_\_\_\_\_

Керівник:

доцент, к. т. н.

Артем ГАБІНЕТ

\_\_\_\_\_

Консультант з нормоконтролю:

Професор, доктор технічних наук

Валерій СІМОНЕНКО

\_\_\_\_\_

Рецензент:

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

Засвідчую, що у цьому дипломному  
проєкті немає запозичень з праць інших  
авторів без відповідних посилань.

Студентка \_\_\_\_\_

Київ – 2020 року

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
ІМ. ІГОРЯ СІКОРСЬКОГО»  
Факультет інформатики та обчислювальної техніки  
Кафедра обчислювальної техніки**

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 123 «Комп'ютерна інженерія»

Освітньо-професійна програма «Комп'ютерні системи та мережі»

ЗАТВЕРДЖУЮ

Завідувач кафедри

Сергій СТИРЕНКО

«\_\_» \_\_\_\_\_ 20\_\_ р.

**ЗАВДАННЯ  
на дипломний проєкт студентці  
Анастасії НЕВМЕРЖИЦЬКІЙ**

1. Тема проєкту «Роботизована рука», керівник проєкту Габінет Артем Вікторович, доцент, к. т. н., затверджені наказом по університету від «07» травня 2020 р. №1081-с

2. Термін подання студентом проєкту 31.05.2020

3. Вихідні дані до проєкту технічна документація

4. Зміст пояснювальної записки

Огляд існуючих рішень, аналіз методів обробки сигналів, проектування, розробка та тестування.

5. Перелік графічного матеріалу

Схема пристрою 1, схема пристрою 2, структурна схема Arduino Uno, алгоритм ЕМД.

## 6. Консультанти розділів проекту

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв
Нормоконтроль	Валерій СИМОНЕНКО		

## 7. Дата видачі завдання 01.09.2020

### КАЛЕНДАРНИЙ ПЛАН

№ п/п	Найменування етапів дипломного проекту	Строк виконання етапів проекту	Примітки
1.	<i>Затвердження теми роботи</i>	01.09.2019	
2.	<i>Вивчення та аналіз завдання</i>	02.09.2019-13.04.2020	
3.	<i>Аналіз методів реалізації</i>	13.04.2020-20.04.2020	
4.	<i>Розробка структури системи</i>	21.04.2020-03.05.2020	
5.	<i>Реалізація системи</i>	04.05.2020-15.05.2020	
6.	<i>Оформлення пояснювальної записки</i>	16.05.2020-25.05.2020	
8.	<i>Передзахист</i>	26.05.2020	
9.	<i>Захист</i>		

Студент

Анастасія НЕВМЕРЖИЦЬКА

Керівник

Артем ГАБІНЕТ

## **Анотація**

Бакалаврський дипломний проєкт присвячений розробці моделі роботизованої руки.

Система розроблена на платформі Arduino з використанням ЕМГ датчиків. Механічна частина проєкту надрукована на 3D принтері. Крім цього використані сервоприводи, ШІМ контролер та модуль бездротової передачі даних.

Розроблена програма для даної системи написана на мові C++ з використанням Arduino IDE. Вона дозволяє ефективно керувати розробленою системою.

## **Annotation**

Bachelor's project is devoted to the development of a model of a robotic hand.

The system is developed on the Arduino platform using EMG sensors. The mechanical part of the project is printed on a 3D printer. In addition, servo drives, PWM controller and wireless data module are used.

The developed program for this system is written in C ++ using Arduino IDE. It allows to effectively manage the developed system.



# Технічне завдання до дипломного проєкту

## ЗМІСТ

1. НАЙМЕНУВАННЯ ТА ОБЛАСТЬ ЗАСТОСУВАННЯ .....	2
2. ПІДСТАВИ ДЛЯ РОЗРОБКИ.....	2
3. МЕТА ТА ПРИЗНАЧЕННЯ РОЗРОБКИ .....	2
4. ДЖЕРЕЛА РОЗРОБКИ .....	2
5. ТЕХНІЧНІ ВИМОГИ.....	2
5.1. Вимоги до розроблюваного продукту .....	2
5.2. Вимоги до програмного забезпечення .....	2
5.3. Вимоги до апаратного забезпечення .....	3

					<i>ДП 6416.02.000 ТЗ</i>		
Зм.	Арк.	№ докум.	Підпис	Дата	<i>Роботизована рука</i> <i>Технічне завдання</i>		
Розроб.		Невмержицька А.Ю.					
Перевір.		Габінет А.В.					
Н. Контр.		Симоненко В.П.					
Затверд.		Габінет А.В.			НТУУ “КПІ”, ФІОТ, каф. ОТ, гр. ІО-64		
					Літ.	Арк.	Аркушів
						1	3

## **1. НАЙМЕНУВАННЯ ТА ОБЛАСТЬ ЗАСТОСУВАННЯ**

Дане технічне завдання розповсюджується на розробку роботизованої руки.

Область застосування: розробка роботизованої руки та створення спеціалізованого програмного забезпечення.

## **2. ПІДСТАВИ ДЛЯ РОЗРОБКИ**

Підставою для розробки служить завдання на розробку роботизованої руки, затвердженою кафедрою обчислювальної техніки Національного технічного Університету України «Київський Політехнічний Інститут ім. Ігоря Сікорського».

## **3. МЕТА ТА ПРИЗНАЧЕННЯ РОЗРОБКИ**

Метою даного проекту є розробка роботизованої руки.

## **4. ДЖЕРЕЛА РОЗРОБКИ**

Джерелами для розробки служать науково-технічна література, публікації в періодичних виданнях, публікації в Інтернеті за даним питанням.

## **5. ТЕХНІЧНІ ВИМОГИ**

### **5.1. Вимоги до розроблюваного продукту**

- Швидкість роботи.
- Надійність роботи, відмово стійкість.
- Можливість вдосконалення.

### **5.2. Вимоги до програмного забезпечення**

- Операційна система MS Windows XP, MS Windows Vista, MS Windows 7, MS Windows 8/8.1, MS Windows 10, MacOSm Linux.
- Arduino IDE

					ДП 6416.02.000 ТЗ	Арк.
						2
Змн.	Арк.	№ докум.	Підпис	Дата		

### 5.3. Вимоги до апаратного забезпечення

- Комп'ютер на базі процесору Intel Dual Core і вище
- Оперативної пам'яті не менше 512 Мбайт

					ДП 6416.02.000 ТЗ	Арк.
						3
Змн.	Арк.	№ докум.	Підпис	Дата		



**Пояснювальна записка**  
**до дипломного проєкту**  
**на тему: «Роботизована рука»**

Київ – 2020 року

# ЗМІСТ

ПЕРЕЛІК ТЕРМІНІВ ТА СКОРОЧЕНЬ.....	4
ВСТУП.....	5
РОЗДІЛ 1. ОГЛЯД ІСНУЮЧИХ РІШЕНЬ.....	6
1.1. Загальні відомості .....	6
1.2 Інтерфейси управління .....	7
1.3. Метод електроміограми.....	9
1.4. Огляд існуючих рішень.....	12
1.4.1. Shadow Hand .....	13
1.4.2. i-Limb.....	14
ВИСНОВКИ ДО РОЗДІЛУ 1 .....	17
РОЗДІЛ 2. АНАЛІЗ МЕТОДІВ ОБРОБКИ СИГНАЛІВ.....	18
2.1. Обробка сигналів.....	18
2.2. Аналіз сигналів.....	25
2.2.1. Метод ЕМД .....	25
2.2.2. Особливості сигналу .....	28
2.2.3. Методи зменшення розмірності отриманих даних .....	31
ВИСНОВКИ ДО РОЗДІЛУ 2 .....	36
РОЗДІЛ 3. ПРОЕКТУВАННЯ.....	37
3.1 Механічна частина .....	37
3.2 Вибір апаратного забезпечення .....	39
3.2.1 Огляд платформи Arduino.....	39
3.2.2. Огляд обраних деталей .....	43
3.3. Вибір програмного забезпечення .....	52
ВИСНОВКИ ДО РОЗДІЛУ 3 .....	53

					<i>ДП 6416.03.000 ПЗ</i>		
Зм.	Арк.	№ докум.	П	Да	<div>Роботизована рука</div> <div>Пояснювальна записка</div>		
Розроб.		Невмержицька А.Ю.					
Перевір.		Габінет А.В.					
Н. Контр.		Симоненко В.П.					
Затверд.		Габінет А.В.			<div>Літ.</div> <div>Арк.</div> <div>Аркушів</div>		
						1	61
					НТУУ “КПІ”, ФІОТ, каф. ОТ, гр. ІО-64		

РОЗДІЛ 4. РОЗРОБКА ТА ТЕСТУВАННЯ .....	54
ВИСНОВКИ ДО РОЗДІЛУ 4 .....	60
ЛІТЕРАТУРА .....	61

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		2

## ПЕРЕЛІК ТЕРМІНІВ ТА СКОРОЧЕНЬ

ЕМГ - електроміографія

АЦП - аналогово-цифровий перетворювач

ЕМД – метод емпіричних мод

МВФ – емпірична мода

ШІМ - широтно-імпульсна модуліція

					ДП 6416.03.000 ПЗ	Арк.
						3
Змн.	Арк.	№ докум.	Підпис	Дата		

## ВСТУП

Сьогодні розвинуті технології використовуються в різноманітних сферах діяльності людини. Однією з провідних наук є робототехніка — прикладна наука, що займається розробкою автоматизованих систем для вирішення різноманітних задач. Робототехніка поєднує у собі основи таких званих STEM дисциплін (природничі науки, технології, інженерію та математику). Область застосування робототехніки в різних сферах дуже широка і не перестає зростати. Робототехніка дозволяє значно зменшити участь людини у багатьох процесах, у небезпечній та важкій праці. Наприклад, праця на складних підприємствах, пов'язаних з хімічною або атомною сферою, вивчення небезпечних територій, проведення рятувальних операцій. Поступово робототехніка з'являється і у повсякденному житті людей. Роботи дозволяють спрости і пришвидшити велику кількість щоденних задач. За прогнозами Міжнародної федерації робототехніки, однієї з найбільших аналітичних організацій в області робототехніки, в наступні роки число роботів зросте майже до 16 млн., і при цьому два з трьох будуть виконувати роботи по дому[1].

Важливим питанням є пошук сфер, пріоритетних для досліджень, прорив у яких серйозно вплине на загальний розвиток робототехніки. Вчені з Інституту Робототехніки при Університеті Карнегі Меллон (США) виділяють шість основних напрямів[2]: структура та шасі роботів, енергозроєнність, електроніка (сенсори), програмне забезпечення (хмарне і бортове), підключення та контроль.

Одним із напрямів розвитку напряму структури та шасі роботів є створення маніпуляторів. Зараз існує багато напрацювань в цій області, але ключовими питаннями є не тільки практичність і надійність маніпуляторів, а й доступність їх для широкого використання. Важливим завданням є створення маніпуляторів, схожих на людську руку, але при цьому при невеликій ціні. Реалізація подібного маніпулятора є метою дипломного проекту.

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		4

# РОЗДІЛ 1. ОГЛЯД ІСНУЮЧИХ РІШЕНЬ

## 1.1. Загальні відомості

Перші дослідження та ідеї робототехніки почали з'являтися у античні часи, задовго до появи самого терміна. Відомими є літальний голуб давньогрецького механіка та інженера Архіта, а Герон Александрійський створив музичну машину, водяний орган і системи рухливих фігур для храмів. Ще один давньогрецький винахідник, Ктесибій, створив водяний годинник, так звану клепсидру. Також описи автоматів є у роботах китайського архітектора Лі Цзи та арабського інженера Аль-Джазарі [3].

Перші андроїди почали з'являтися близько 30-х років XX століття, вони були простими і виконували елементарні рухи. Швидкий розвиток робототехніки спостерігався по закінченню Другої світової війни. Вже у 1948 році, американський математик Норберт Вінер сформував закони кібернетики. У 50-х роках були розроблені перші механічні маніпулятори, вони використовувались для роботи з радіоактивними матеріалами. Маніпулятори копіювали рухи оператора, який знаходився на безпечній відстані. Широке поширення маніпулятори набули вже у 60-х роках. Пізніше, з появою дешевших мікропроцесорних систем керування, зменшилась вартість роботів, що було стимулом для їх широкого поширення по більшості сфер промисловості [4].

Розвиток робототехнічних рук є досить актуальним в останні роки, завдяки широкому використанню: протезування кінцівок та автоматизація багатьох виробничих завдань. Для роботи з будь-якими предметами та інструментами, доцільно розробляти модель з дизайном людської руки. Саме тому антропоморфні маніпулятори є метою для розробки.

Як зазначено у вступі — тема взаємодії людини і робота, а саме контроль і управління, залишається важливою проблемою робототехніки. Інтерфейс людина-робот має провідну роль через те, що використання роботів стає все більш поширеним і переходить на повсякденні задачі.

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		5

Велика кількість інтерфейсів управління доволі складна. Вони можуть містити складні датчики та механізми, а користувач повинен точно виконувати потрібні дії і вміти відобразити їх на рухи роботів.

Дослідження спрямовані на розробку нових методів передачі інформації між людиною та роботом, спрощення їх взаємодії через інтуїтивно зрозумілий інтерфейс.

В сучасній робототехніці є два основних підходи до управління: телеуправління та автономність. Телеуправління залишає керування людині, а іноді і цілій групі людей. Автономність навпаки прагне зробити робота цілком автономним на всіх етапах. Цей підход є більш складним і часто потребує залучення штучного інтелекту або нейромереж. Роль людини при цьому змінюється від оператора до супервайзера.

## 1.2 Інтерфейси управління

По типу управління робототехнічні системи поділяються на [5]:

1. Біотехнічні – управляються людиною.

- Командні – виконання вмикається з пульта управління дистанційно.
- Копіюючи – управління здійснюється пристроєм, подібним до виконавчого пристрою робота і розташованим на певній відстані.
- Напіваавтоматичні – натискаючи керуючі елементи з деяким ступенем свободи, оператор задає необхідні переміщення роботу.

2. Автоматичні – процес управління відбувається без безпосередньої участі людини.

- Програмні – працюють за спроектованою програмою, що збережена у пам'яті і має жорстку послідовність дій.
- Адаптивні – мають датчики, що дозволяє виконувати операції у неповністю визначеній обстановці з адаптацією до неї.
- Інтелектуальні – мають елементи штучного інтелекту, вони здатні розпізнавати предмети, проробляти план вирішення задач, змінювати дії при певних обставинах.

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		6

### 3. Інтерактивні

- Автоматизовані – чергує біотехнічні и автоматичні режим.
- Супервизорні – частини циклу виконуються автоматично, а переходи задає оператор.
- Діалогові – передбачає автоматичне виконання задач з спілкуванням с оператором, робот бере участь у формуванні задач.

Можна виділити кілька інсуючих інтерфейсів, які підійдуть для управління маніпулятором [6]:

1. Аватар-костюми добре підходять для управління роботами андроїдного типу. Костюм аналізує рухи людини і інтерпретує їх у команди для робота. У таких костюмах часто використовуються датчики нахилу, положення та дотику.
2. Захоплення рухів реалізується різними методами, але самим поширеним є використання камери, що аналізує зміну положення людини і передає їх як команду для виконуючого пристрою.
3. Розмовний інтерфейс розрізняє команди оператора, подані голосом. Система розпізнає і інтерпретує команди в управляючі сигнали.
4. Нейроінтерфейси використовують сигнали, що надходять безпосередньо від людини. Вони дозволяють знімати сигнали з мозку чи м'язів людини і інтерпретувати їх в команди для пристроїв.

Цікаві приклади реалізації маніпуляторів є у сфері медицини. Сучасні технології і інтерфейси часто використовуються у протезуванні. Розвиток науки та техніки дозволяє розробити протези, що використовують нейроінтерфейси для управління протезом-маніпулятором. Основними способами управління є нейрокомп'ютерний, нейроелектричний, електроміографічний.

У нейрокомп'ютерному методі використовується інформація, що зчитується з підкорки головного мозку. Сигнал знімається за допомогою приладів, що вживлені в кору мозку. Отримані дані передаються механічну частину маніпулятора. Великим недоліком цього методу є інвазивність та пов'язана з цим можливість зараження та ушкодження мозку у процесі вживлення електродів та експлуатації.

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		7



Нейроелектричний метод використовує сигнали, що знімаються з активних рухових нервів людини. Електроди можуть бути вживлені безпосередньо у нервову тканину, або можуть бути встановлені над нервами. Недоліками методу є інвазивність та можливість враження нерву.

Для зчитування сигналів м'язевої активності використовується метод електроміографії, який полягає у реєстрації електричних потенціалів м'язів. У стані спокою біоелектрична активність відсутня, а при скороченнях виникають коливання. Цей метод є неінвазивним, тому може широко використовуватись без складних операцій та вживлення датчиків. Цей метод обрано для управління маніпулятором [7].

### 1.3. Метод електроміограми

Методи електроміограми досить поширені у протезуванні кінцівок. Найбільш оптимальними є біоелектричні протези, які дозволяють використовувати фізіологічні сигнали для управління рухами кінцівок. Електроміографічний метод дозволяє створити прості, природні інтерфейси управління. Цей інтерфейс забезпечує швидкий і практичний спосіб передачі команд, і дозволяє в реальному часі керувати потрібними пристроями.

Електроди, що розміщені безпосередньо на шкірі користувача, реєструють м'язеву активність, яка перетворюється у кінематичні змінні, що у свою чергу використовуються для управління пристроєм. Подібність пристроїв до людини мінімізує складність управління, так як природних рухів буде достатньо для управління.

Для запису сигналів електроди потрібно розміщувати на конкретних місцях, які доволі легко знайти. Сучасні прилади дозволяють обробляти сигнал у електроді і бездротово передавати сигнал на станцію обробки, що підвищує практичність і ефективність запису.

Електричну активність м'язів можна розділити на спонтанну та активність, викликану внутрішнім імпульсом або зумовлену штучним імпульсним зовнішнім впливом. Зовнішній вплив може бути механічним або електричним, наприклад, удар молоточком по сухожиллям м'язів.

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		8

Електроміограма є кривою, яка реєструє електричну активність м'язів. Для цього застосовується спеціальний прилад - електроміограф, який реєструє зміну в часу різниці потенціалів [8].

На одиночний вплив м'яз реагує одиночним скороченням. При цьому можна виділити три фази:

1. латентний період (від 2-3 до 10 мс), що триває від моменту нанесення роздратування до початку скорочення;
2. фаза скорочення (40-50мс);
3. фаза розслаблення (близько 50 мс).

Можна виділити наступні види електроміографії:

1. електроміографія за допомогою голчастих електродів, які вводяться в м'яз;
2. електроміографія з використанням нашкірних електродів;
3. стимуляційна електроміографія.

У першому випадку електроди реєструють зміни потенціалу в окремих волокнах або в групі волокон. При використанні нашкірних електродів, буде зареєстрована сумарна електроміограма. Стимуляційні методи застосовуються в електроміографії досить часто. При цьому за рахунок нашкірних електродів накладених на рухові точки проводиться стимуляція м'язів з подальшою реєстрацією та аналізом відгуку на цей вплив. Тип необхідного електрода вибирається в залежності від цілей проведених експериментів.

У більшості м'язів ЕМГ активність передуює руху, що дозволяє передавати інформацію про наміри руху до того, як рух здійсниться. Це дозволяє мінімізувати затримку у системах, де такі затримки відіграють важливу роль. ЕМГ система може давати інформацію про опір м'язів без механічного впливу, а також про втому, з якою може бути пов'язаний загальний стан людини.

ЕМГ сигнали є інформативними, але вони мають і деякі недоліки. Сигнал ЕМГ досить складно аналізувати візуально, так як він має шумовий характер з широким спектром. Тому методи комп'ютерної обробки сигналів набули широкого поширення. Для реєстрації сигналів ЕМГ доводиться використовувати досить розширений діапазон частот: верхня смуга цього діапазону - 10-20 кГц, а нижня обмежена на рівні - 2-10 Гц. Для прикладу, м'яз-

					ДП 6416.03.000 ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

згинач пальця людини має характерну частоту - 113 Гц. Максимальна частота синхронного відгуку при подразненні окремих рухових нервових волокон жаби становить - 300 Гц, оптимальною є частота - 75-150 Гц; м'язові волокна - 150 Гц, оптимальна - 20-50 Гц.

У електроміограмі міститься важлива інформація не тільки про стани окремого м'яза, але також і про стан центральної нервової системи (спинний і головний мозок), яка управляє всіма м'язами. це пов'язане з тим, що посилаючи м'язам сигнали, мозок управляє руховими функціями.

При розробці таких біологічно керованих маніпуляторів основним завданням є необхідність визначення методу отримання даних при фантомному переміщенні пацієнта, під час якого імплантовані або поверхневі електроди формують за рахунок електричної активності м'язів імпульси, що подаються на підсилювач, що дозволяє отримати характеристики і вид руху. Такі величини обробляються для перетворення в необхідні сигнали управління для здійснення руху виконавчого механізму штучної кінцівки. Очевидно, що для створення якісних маніпуляторів з використанням біологічного або нейроелектричного управління необхідно мати якісний і стабільний сигнал. Тому першочерговим завданням при розробці таких протезів є розробка підсилювача біопотенціалів, визначення надійного пристрою електродів і методу їх накладення, а також знаходження необхідних місць кріплення або імплантації. Слід врахувати, що вид корисного сигналу може спотворюватися за рахунок накладення шумів, амплітуда яких зазвичай істотно перевершує сам ЕМГ-сигнал. При наявності великої відстані між електродами збільшується можливість виникнення шумів в корисному сигналі, які пов'язані з генерацією шумів в сполучних проводах. Наступна серйозна складність виникає за рахунок появи перехресної перешкоди від сусідньої групи м'язів. Таким чином, основним завданням в цьому випадку є максимальне ослаблення взаємного впливу електричної активності сусідніх м'язів. Також на маніпулятор впливає навколишнє середовище: наявність поту на шкірі призводить до погіршення її контакту з електродами, також можлива поява механічного впливу на електроди. Крім того, слід брати до уваги ефект поляризації електродів, наявність високого імпедансу шкіри, надійність

					ДП 6416.03.000 ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

кріплення електродів і ін. Внаслідок того, що амплітуда ЕМГ сигналу відрізняється у різних людей і для різних видів м'язів, тож доводиться вибирати коефіцієнти підсилювача в досить широких межах. Точна реєстрація потенціалу здійснюється за рахунок безпосереднього контакту датчика з тілом. У цьому випадку між датчиком і шкірою вводиться електропровідний гель. Хоча наявність гелю призводить до значному поліпшенню якості контакту, це може привести до недоліків. Гель може поширитися по поверхні тіла, внаслідок чого утворюється коротке замикання між датчиками. Наступним недоліком застосування гелю є неможливість проводити довготривалі вимірювання, так як гель може висихати.

М'язи зафіксовані в шарах інших тканин, тому сигнали можуть змінюватись з різних причин, наприклад втома, зміна положення електрода або піт на місці запису.

Кістково-м'язева система людини є достатньо складною, а зв'язок між сигналами та рухами доволі нелінійний, тому в більшості випадків реалізується контроль, який фокусується тільки на деяких елементах управління. Наприклад, контроль над зап'ястям, плечем або пальцями. Більшість робот контролює тільки кінцеві положення. Таким чином, управління за допомогою ЕМГ сигналів може викликати багато проблем з плавністю управління. Для рішення подібних проблем часто використовуються методи машинного навчання.

#### 1.4. Огляд існуючих рішень

В світі існує багато маніпуляторів, подібних до того, що є розроблюється. Перші моделі маніпуляторів, які використовують міоелектричні датчики, почали з'являтися ще у 1980-х роках. У цих моделях використовувався лише один двигун, а рухались великий, вказівний та середній пальці. Принцип захвата був схожий на кліщі. Наприкінці 80-х років, проекти базувались на підході, який полягає у зменшенні кількості приводів та зменшенні складності без впливу на універсальність.

З роками принцип роботи був значно покращений, кількість двигунів була збільшена, у більшості девайсів використовується по одному двигуну на кожен

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

палець. За останні 10 років проведено багато вдосконалень, що пов'язані з сенсорами, механічною частиною, дизайном та контролем. Моделі стали схожими на людську руку та вміють виконувати різні задачі. Сьогодні існують велика кількість різноманітних проектів таких маніпуляторів, які використовуються у різних сферах.

#### 1.4.1. Shadow Hand



Рисунок 1.1 - Shadow Hand

Shadow Dexterous Hand – це роботизована рука, подібна до людської, яка була розроблена компанією The Shadow Robot Company в Лондоні. Рука аналогічна людській за розміром та формою, вона має 24 сугави та 20 ступенів свободи, Проект був розроблений так, щоб мати схожий до реальної руки діапазон рухів. Зараз існує два різних варіанти моделі: електрична та пневматична. Електрична рука виконує рухи за допомогою двигунів постійного струму, а пневматична — за допомогою пневматичних штучних м'язів. Пневматичні штучні м'язи (ПШМ) — це пристрої, які управляються стислим повітрям, яке наповнює пневматичний пузир.

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

Рука була створена для вивчення захвату, маніпуляції, нервового контролю, нейроінтерфейсу та промислового керування якістю і маніпуляцій з небезпечними матеріалами. Маніпулятор широко використовується в багатьох установах, зокрема у NASA, в різних університетах та дослідницьких проектах.

Рука є самодостатньою, всі системи вбудовані в руку та передпліччя [9].

#### 1.4.2. i-Limb



Рисунок 1.2 - i-Limb Pulse

i-Limb – перший у світі комерційний проект біонічного протезу. Розроблений у Едінбурзі, проект був запусканий у Ванкувері ще у 2007 році. Рука керується ЕМГ сигналами, які знімаються з двох електродів, що розміщені на певних м'язах. При скороченні цих м'язів, сигнали від електродів поступають на мікропроцесор, який керує механічною частиною руки і рухами. Сигнали налаштовані на 4 режими. У останніх розроблених моделях навіть є функція керування зі смартфона. Девайс має 11 сугавів і 6 ступенів свободи, масу приблизно 450-600 г. Механічна частина керується двигунами постійного струму. Модель руки пропонує вибір автоматизованих жестів та можливості налаштування програмного забезпечення. Автоматизовані елементи забезпечують велику точність виконання завдання.

Найсерйознішим недоліком цього проекту є дуже висока ціна — від 17 тис. Долларів США [10].

### 1.4.3. Bebionic



Рисунок 1.3 - bebionic hand

BeBionic hand – проект, розроблений у Лейпціну, вперше представлений у 2010 році. Проект є одним із найдосконаліших міоелектричних протезів, використовує передові технології та ергомічні функції. Рука керується ЕМГ сигналами, датчики розміщені в гнізді моделі. Інструкції обробляються та пересилаються на 337 механічних деталей, які імітують рухи реальної руки. Модель має 11 сугавів та 6 ступенів свободи. Маса руки — близько 500 г. Механічна частина керується 5 двигунами постійного струму, по одному на кожен палець. Рука здатна обробляти 14 різних зчеплень та положень руки. Є функція бездротового управління і можливість нашатування програмного забезпечення.

Недолік є таким самим, як у попереднього пристрою. Ціна такого маніпулятора — 11 тис. Долларів США [11].

Крім цього, цікавим є розробка компанії Microsoft ЕМГ кільця. Воно включає у себе 10 електродів, що посиляють бездротові сигнали на комп'ютер.

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

Було розроблено додаток, що дозволяє користувачу використовувати інтерфейс м'язи комп'ютер для гри в популярну ком'ютерну гру Guitar Hero. В цій грі у користувача є контролер подібний до гітари, але з кнопками замість струн, а система дозволяє ЕМГ дозволяє імітувати рухи, як при грі на реальній гітарі. Подібна система була розроблена для гри у Super Mario. Були запрограмовані спеціальні жести, яки імітували натискання на потрібні кнопки.

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		15



## ВИСНОВКИ ДО РОЗДІЛУ 1

В першому розділі було розглянуто важливі питання розвитку робототехніки, одним із яких є створення маніпуляторів, подібних до людської руки. Однією з проблем реалізації такого проекту є вибір інтерфейсу управління маніпулятором. Після вивчення існуючих варіантів управління, було обрано нейроінтерфейси, а саме метод електроміографії. Цей метод дозволяє знімати сигнали управління безпосередньо з м'язів людини за допомогою нашкірних електродів. При цьому ЕМГ сигнали є складними для аналізу, потрібно враховувати шуми та вплив сусідніх м'язів, тому важливою є попередня обробка знятих сигналів.

Зараз існує багато проектів, що реалізують маніпулятори, які можуть використовуватись у різних сферах та проектах. Після аналізу існуючих реалізацій маніпулятора, виявлено такі важливі деталі:

1. Моделі мають велику кількість сугавів та степенів свободи
2. Кожен палець керується окремим двигуном або іншим пристроєм
3. Вага близька до реальної ваги людської руки

Недоліками існуючих проектів є дуже велика ціна. Тому вирішено розробити схожий за функціоналом маніпулятор, який при цьому буде значно дешевшим у реалізації і відповідно більш доступним.

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16

## РОЗДІЛ 2. АНАЛІЗ МЕТОДІВ ОБРОБКИ СИГНАЛІВ

У першому розділі було проведено аналіз існуючих методів управління робототехнікою і обрано метод електроміографії. Існує декілька розробок ЕМГ датчиків, що считують активність м'язів. Їх принцип роботи пов'язаний з обробкою отриманих від м'язів сигналів.

### 2.1. Обробка сигналів

Сигнал це фізична величина, що містить інформацію про зміну деяких змінних. Математично сигнал виражається як функція однієї або декількох змінних. Сигнали можуть з'являтися через природні або технічні процеси [12].

Сигнали можуть бути класифіковані кількома способами. За фізичною природою вони поділяються на [13]:

- механічні – сигнал у вигляді дії твердого тіла, дієвою величиною якого є переміщення, сила або момент сили;
- електричні – сигнал у вигляді електричної дії, дієвою величиною якого є струм або напруга;
- радіосигнал – сигнал у вигляді дії електромагнітного випромінювання, дієвою величиною якого є напруженість електричного поля або магнітного поля;
- оптичний — сигнал у вигляді дії оптичного випромінювання, дієвою величиною якого є потік випромінювання;
- акустичний – сигнал у вигляді дії звуку, дієвою величиною якого є звуковий тиск;
- гідравлічний – сигнал у вигляді механічної дії рідини (газу), дієвою величиною якого є тиск.

Для електроміограми використовується так званий біологічний сигнал. Біосигнал може бути визначений, як фізичне явище, при якому фізичні або хімічні величини характеризують біологічний стан людини.

Залежно від функції, яка описує сигнал, вони поділяються на:

- неперервні (аналогові), що описуються неперервною функцією;
- дискретні, що описуються функцією відліків, взятих в певні моменти часу;

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		17

- квантовані за рівнем;
- дискретні сигнали, квантовані за рівнем.

Найбільша кількість з біологічних сигналів – неперервні. За допомогою техніки виборки їх можна перетворити у дискретні сигнали.

Електрична активність в організмі необхідна для підтримання життя. Джерело цієї активності є на клітинному рівні (наприклад нейрони). Активність є результатом руху заряджених йонів.

Через електрохімічну активність клітин утворюються біоелектричні потенціали. Вони є компонентами м'язової, залозистої або нервової тканини. Коли клітини стимулюються належним чином, вони виявляють потенціал дії, в іншому випадку вони залишаються або коливаються навколо їх потенціалу спокою.

Електроміографія (міоелектрична активність) є результатом запису електричної активності м'язів.

Існує 2 види запису сигналу ЕМГ:

1. Внутрішньом'язовий
2. Поверхневий

У кожному випадку електрод або уніполярний або біполярний. Записаний сигнал має низьку інтенсивність і містить достатньо шуму. Для його обробки та використання потрібно посилення (посилений лінійно по всьому діапазону підсилювача і записуючої системи)

На відміну від дискретного, який містить певну кількість інформації, аналоговий сигнал містить нескінчену інформацію і є частиною аналога. Як було зазначено вище, біологічні сигнали є аналоговими, тому їх потрібно перетворити у цифрові для обробки (аналогово-цифрове перетворення), так як комп'ютер вміє обробляти виключно цифрові сигнали.

Перетворення сигналу з аналогового у цифровий проходить у 3 етапи: дискретизація, квантування і кодування.

Структурна схема перетворення сигналу зображено на рисунку 2.1.



Рисунок 2.1 – структурна схема перетворення сигналу

При цьому використовується АЦП – аналогово-цифровий перетворювач. АЦП – це пристрій, що виконує перетворення вхідної величини в її числове представлення. Формально, вхідною величиною може бути будь-яка фізична величина. Наприклад, це може бути напруга, струм, опір та інше.

Схема АЦП зображена на рисунку 2.2.

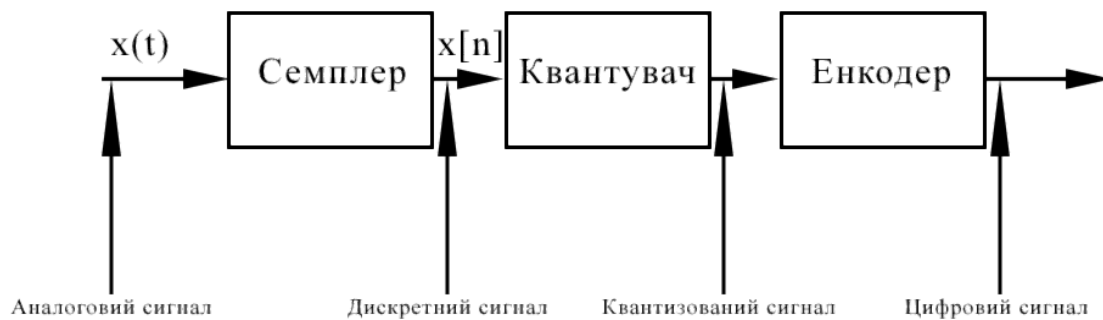


Рисунок 2.2 – схема АЦП

Семплер є системою, яка виконує процес дискретизації і витягує вибірки від безперервного сигналу. Теоретично ідеальний семплер виробляє вибірки, які відповідають миттєвим.

Дискретизація це перший крок перетворення сигналу з аналогового у цифровий. На цьому етапі с деяким періодом дискретизації беруться сигнали. Відповідно, частота дискретизації є оберненою до періоду. Для зменшення динамічних помилок перетворення, зумовлених зміною сигналу за час відліку, потрібно щоб процес отримання відліку вхідного сигналу займав дуже малий проміжок часу. Частота дискретизації може бути отримана с теореми Котельнікова. В ній стверджується, що для того щоб по відліку сигналу можна було якомога точно відновити неперервний сигнал, необхідно щоб частота дискретизації перевищувала верхню частоту спектра дискретизуємого сигналу не менш як у 2 рази.

Будь який сигнал має спектральне представлення. Будь яку представлення сигналу – це представлення суми його гармонійних складових різних частот взятих з деякими ваговими коефіцієнтами. Для неперіодичних сигналів це інтеграл, а для періодичних – сума. Перехід до сигналу від спектра виконується за допомогою перетворення Фурьє.

Як приклад, розглянемо перехід у вигляді періодичної функції, графік якої зображено на рисунку 2.3.

$$L(t) = L(t + KT) \quad (2.1)$$

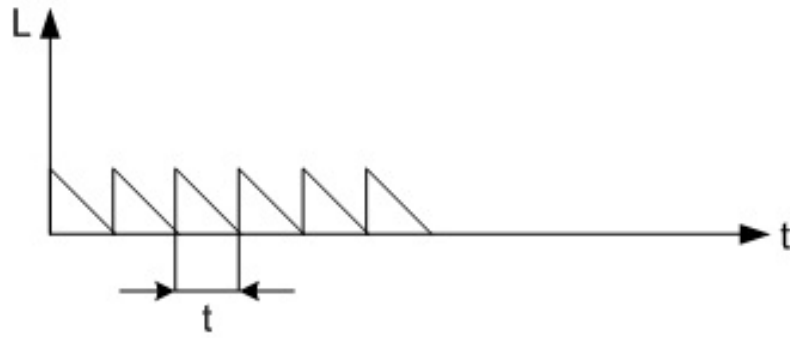


Рисунок 2.3 – графік функції  $L(t)$

Періодична функція може бути представлена у вигляді ряду гармонійних функцій.

$$\sum_{-\infty}^{\infty} C_n e^{\frac{i2\pi n}{T}t} \quad (2.2)$$

Далі, по формулі Ейлера, будь-який вираз можна представити у вигляді

$$e^{ix} = \cos x + i \sin x, \quad (2.3)$$

$\frac{1}{T} = f_1$  – частота першої гармоніки;

$\frac{n}{T} = nf_1 = f_n$  – частота  $n$  – ої гармоніки;

$2\pi f_n = \omega_n$  – кутова частота  $n$  – ої гармоніки;

$C_n = |C_n|e^{-i\phi_n}$  – комплексна амплітуда гармоніки,

де  $\phi_n$  – фазовий спектр.

Сукупність амплітуд гармонік ряду Фур'є називається амплітудним спектром, а сукупність фаз відповідно називається фазовим спектром.

$$C_n = \frac{1}{T} \int_{-\frac{T}{2}}^{\frac{T}{2}} L(t) e^{-\frac{i2\pi n}{T}t} dt \quad (2.4)$$

Приклад спектра зображений на рисунку 2.4.

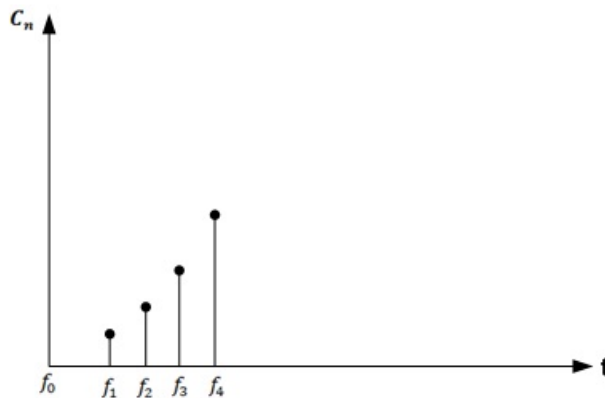


Рисунок 2.4 – приклад спектра

Для неперіодичної функції  $T \rightarrow \infty$ , а  $f \rightarrow 0$ , тоді  $f_n = \frac{n}{T}$  замінюється неперервно змінною частотою, а сума замінюється інтегралом.

$$L(t) = \frac{1}{2\pi} M(\omega) e^{i\omega t} d\omega \quad (2.5)$$

Пряме перетворення Фур'є для неперервного сигналу

$$M(\omega) = \int_{-\infty}^{\infty} L(t) e^{-i\omega t} dt \quad (2.6)$$

Таким чином спектр неперіодичної функції є сумою нескінченної кількості гармонійних коливань, частоти яких знаходяться нескінченно близько один до одного.

Квантування це процес перетворення вибірок, отриманих з виходу семплера, в випадкову послідовність дискретних значень, які належать кінцевому набору рівнів амплітуд. Значення, які може приймати цифровий сигнал, називаються рівні квантування.

Кількість рівнів квантування можна знайти за формулою:

$$N = 2^n, \quad (2.7)$$

де  $n$  – кількість розрядів;

$N$  – рівень квантування.

Вибір кількості рівнів квантування враховує необхідність точного представлення сигналу, що потребує велику кількість рівнів квантування, а з іншої сторони – зменшення кількості рівнів для зменшення розрядності коду.

На рисунку 2.5 представлений процес квантування.

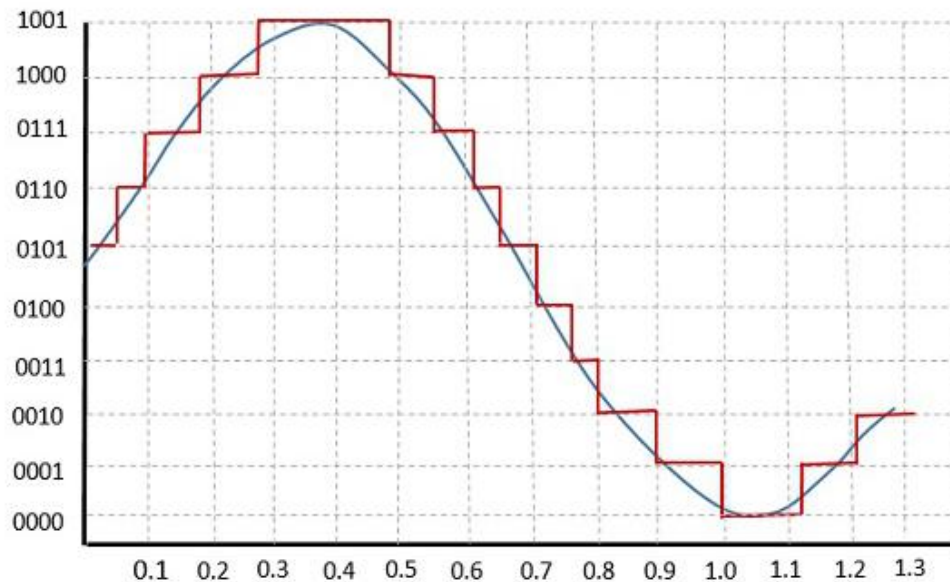


Рисунок 2.5 – квантування сигналу

Існують два способи квантування: рівномірне і нерівномірне квантування. При рівномірному квантуванні рівні квантування розташовані однорідно. При неоднорідному квантуванні рівні розташовані неоднорідно, і відношення між ними є логарифмічним.

На відміну від аналогового сигналу, отриманий квантований сигнал може приймати кінцеву кількість значень. Це дає змогу представити його в межах кожного інтервалу дискретизації певним числом, що рівний порядковому номеру квантування. Отримане число можна виразити комбінацією деяких чисел та символів. Отриманий вираз називається кодовим словом. Квантований сигнал можна представити у вигляді таких виразів. Отже, кодування це перетворення квантованих вибірок в відповідну форму, що означає, що кожне значення квантування вибірок представлено числом, що складається з  $n$  бітів. Якщо квантований сигнал приймає  $N$  значень, то число символів в кожному кодовому слові дорівнює  $n \geq \ln N$ . Отримані кодові слова можна передавати у 2 формах: паралельній та послідовній. Для передачі паралельним методом використовується декілька ліній зв'язку, сигнали при цьому передаються одночасно. Для передачі в послідовній формі інтервали дискретизації ділять на такти і сигнали передаються послідовно впродовж такту.



## 2.2. Аналіз сигналів

### 2.2.1. Метод ЕМД

Сигнал ЕМГ, пов'язаний з рухами руки, можна вважати нестационарним та нелінійним. Таким чином, використання методів розкладання для стаціонарних та лінійних процесів забезпечить некоректні результати. З цієї причини використання методу ЕМД [14] є кращим для даного випадку, оскільки є він адаптивним. Метод ідеально підходить для нестационарних та нелінійних сигналів.

ЕМД – це метод розкладання сигналів на функції, що отримали назву емпіричних мод. Цей метод представляє собою ітераційну процедуру, в результаті якої вхідний сигнал розкладається на емпіричні моди або внутрішні коливання.

Метод ЕМД спирається на такі припущення:

1. в сигналі є щонайменше два піки (одне максимальне і одне мінімальне значення),
2. часовий інтервал між піками визначає репрезентативну міру часу.

У випадку, коли немає піків, але існують точки перегину, екстремуми будуть отримані після диференціювання початкового сигналу один або декілька разів. Результат буде отриманий за рахунок інтеграції компонентів.

ЕМД є важливою частиною трансформації Гільберта-Хуанга. Цей метод може розбити будь-який складний набір даних на декілька кінцевих компонентів, названих функціями внутрішніх мод. Кожна функція зображує просте коливання мод, яке доповнює просту гармонічну функцію. Обмеження, встановлені у визначенні функцій, забезпечують добре перетворення Гільберта. Метод ЕМД виявився високоефективним та адаптивним, що робить його ідеальним для нестационарного та нелінійного процесу [15].

Для виконання функції, необхідно виконати дві умови:

1. Кількість піків повинна бути рівною або різнитися здебільшого на одиницю від кількості нульових перетинів у повній формі хвилі

					ДП 6416.03.000 ПЗ	Арк.
						24
Змн.	Арк.	№ докум.	Підпис	Дата		

2. Середнє значення обох огибаючих сигнала на основі локальних максимумів і мінімумів дорівнює нулю [14].

Ідея ЕМД полягає в реалізації коливальних сигналів на рівні їх локальних коливань і формуванні поняття, що сигнал - це суперпозиція швидких коливань на повільних, як це видно на рисунку 2.6.

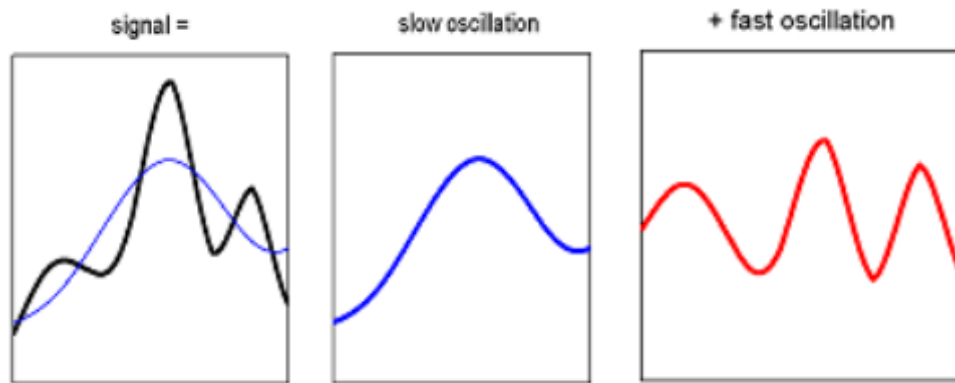


Рисунок 2.6 – сигнал як суперпозиція швидких та повільних коливань

Сигнал  $x(t)$  розкладається за допомогою метода ЕМД за алгоритмом:

1. Усі локальні екстремуми  $x(t)$  містять дві огибаючі, одна для локального максимуму ( $e_{max}(t)$ ) і одна для локального мінімуму ( $e_{min}(t)$ ), знайдених за допомогою інтерполяції.
2. Середнє значення оцінюється за допомогою огибаючих, знайдених у попередньому кроці:  $m(t) = \frac{e_{min}(t) + e_{max}(t)}{2}$ .
3. Середнє значення віднімається від сигналу, створює детальний сигнал,  $d(t) = x(t) - m(t)$ .
4. Кроки повторюються з заміною  $x(t)$  на  $m(t)$  до досягнення монотонної функції, яка буде остаточною залишковою (або досягнуто певна кількість МВФ, яка встановлена користувачем).

Третій крок не обов'язково створюватиме МВФ на основі раніше встановлених обмежень. Таким чином, сигнал деталізації,  $d(t)$  буде відповідати цим критеріям після просіювання [14] [16]. Сумою МВФ та залишкового члена можна відновити початковий сигнал  $x(t)$ :

$$x(t) = \sum_i IMF_i + r(t) \quad (2.8)$$

де  $IMFi$  можна вважати частиною високої частоти, а залишок частиною низької частоти. Схема алгоритма подана у додатку Г.

Після використання алгоритму ЕМД можна використовувати перетворення Гільберта для кожного МВФ для обчислення миттєвої частоти, яка є похідною фазової функції. Перетворення Гільберта – лінійне інтегральне перетворення, що ставить у відповідність однієї функції іншу в тій самій області. Таким чином, перетворення Гільберта оцінює амплітуду і частоту кожного МВФ як функцію часу. Початковий сигнал також може бути виражений як реальна частина перетворення Гільберта [17], використовуючи таке рівняння:

$$x(t) = PR(\sum_i a_j(t) \cdot e^{i\theta_j(t)} = PR(\sum_j a_j(t) \cdot e^{i \int \omega_j(t) dt} \quad (2.9)$$

Спектр Гільберта-Хуана ( $H(\omega, t)$ ) - це амплітудний частотно-часовий розподіл. Метод ЕМД можна зрозуміти по рисунку 2.7, 2.8, що показує покроковий процес створення МВФ.

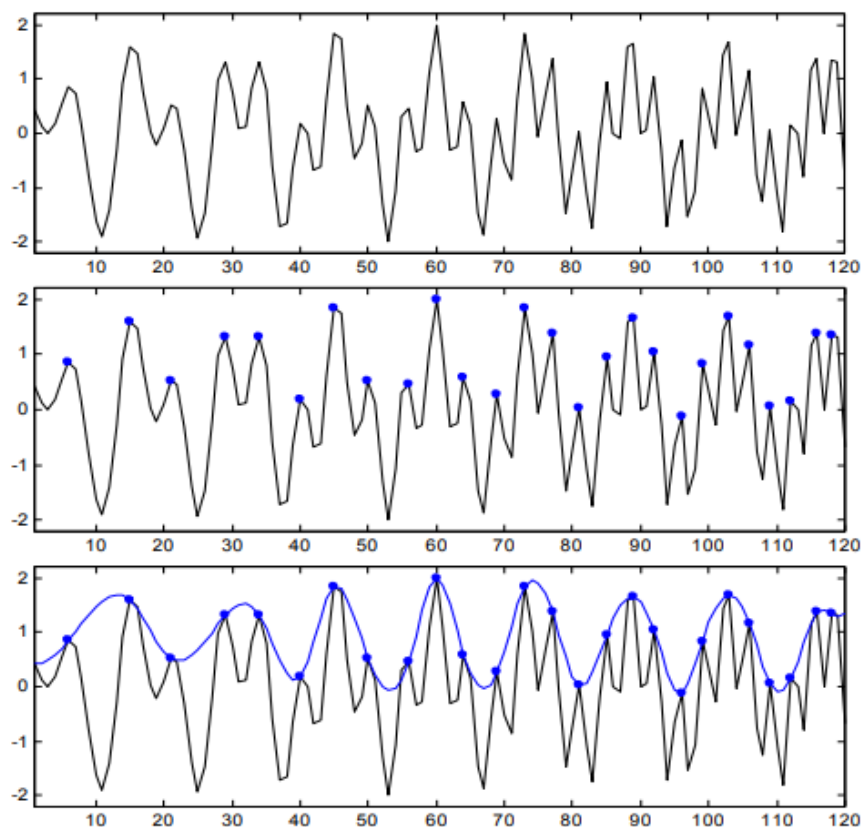


Рисунок 2.7 – створення МВФ (1)

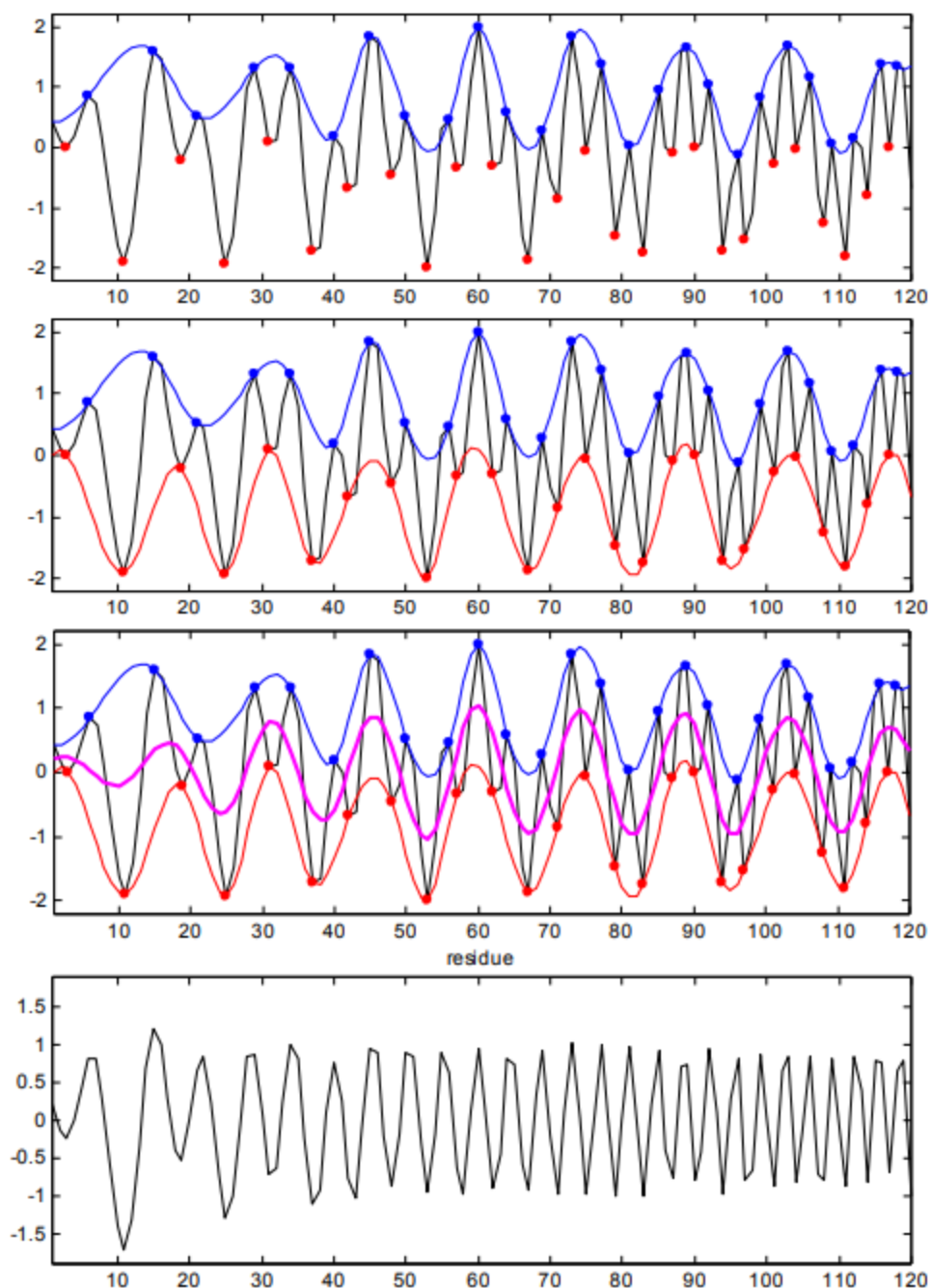


Рисунок 2.8 – створення МВФ (2)

### 2.2.2. Особливості сигналу

Отриманий необроблений сигнал в більшості випадків не має використання. Але навіть у такому випадку, існує деяка інформація, яку можна виділити з необробленого сигналу. Функції виділення інформації проводять стиснення інформації та вирішують проблеми розмірності, що вирішуються методами, які перераховані далі.

Дані, які будуть отримані з урахуванням особливостей кожного сигналу, можна використовувати в матрицях і як вхідні данні для алгоритмів. Наприклад, такі алгоритми - це методи розпізнавання образів.

### 2.2.2.1. Особливості часової області

Можна виділити такі особливості часової області[18]:

1. Середнє значення: сума значень вибірок сигналу, поділена на кількість вибірок. Якщо сигнал приймає від'ємне значення, доволі часто береться абсолютне значення сигналу.

Математичний вираз:

$$\mu = \frac{1}{N} \sum_{k=1}^N |x_k| \quad (2.10)$$

де  $x_k$  це  $k$ -те значення з  $N$  вибірки сигналу.

2. Дисперсія: це міра щільності потужності сигналу, а його квадратний корінь називається стандартним відхиленням ( $\sigma$ ).

Дисперсія обчислюється за формулою:

$$VAR = \frac{1}{N-1} \sum_{k=1}^N x_k^2 \quad (2.11)$$

3. Перетин нуля: підраховує кількість разів, через які сигнал проходить через нуль або змінює знак, і, таким чином, виражає частоту. Потрібно встановити поріг для зменшення шуму, викликаного при нульовому перетині.

Використовуючи два суміжні зразки сигналу ЕМГ  $x_k$  та  $x_{k+1}$  перетин можна обчислити таким чином:

$$ZC = \sum f(x), \quad f(x) = \begin{cases} 1, \text{ якщо } (x_k > 0, x_{k+1} < 0) \\ \text{або } (x_k < 0, x_{k+1} > 0) \\ 0 \end{cases} \quad (2.12)$$

для  $k = 1, 2, 3, \dots, (N - 1)$

4. Зміни знаку нахилу: міра часу, коли нахил сигналу змінює знак. Для трьох послідовних вибірок сигналу  $x_{k-1}$ ,  $x_k$  та  $x_{k+1}$ , зміну знаку можна обчислити так:

$$SSC = \sum f(x), \quad f(x) = \begin{cases} 1, \text{ якщо } (x_k < x_{k+1}, x_k < x_{k-1}) \\ \text{або } (x_k > x_{k+1}, x_k > x_{k-1}) \\ 0 \end{cases} \quad (2.13)$$

для  $k = 1, 2, 3, \dots, (N - 1)$

					ДП 6416.03.000 ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

5. Довжина хвилі: це міра сукупної зміни сигналу. Для двох сусідніх зразків сигналу ЕМГ  $x_k$  та  $x_{k+1}$ , вона визначається як:

$$WL = \sum_{k=1}^{N-1} (|x_{k+1} - x_k|) \quad (2.14)$$

6. Амплітуда Віллісона: це кількість разів, коли дві послідовні вибірки перевищують конкретний поріг. З урахуванням двох суміжних зразків сигналу ЕМГ  $x_k$  та  $x_{k+1}$ , амплітуда обчислюється за формулою:

$$WAMP = \sum_{k=1}^{N-1} f(|x_{k+1} - x_k|), \quad f(x) = \begin{cases} 1, & \text{якщо } x > \text{порогу} \\ 0 & \end{cases} \quad (2.15)$$

7. Коефіцієнт ексцесу: міра розподілу ймовірності дійсної випадкової величини [10]. Характеризує стрімкість підвищення кривої розподілу, порівняно з нормальною кривою. Обчислюється за формулою:

$$k = \frac{E(x-\mu)^4}{\sigma^4} \quad (2.16)$$

де  $E$ - очікуване значення,  $\sigma$  - стандартне відхилення.

8. Коефіцієнт асиметрії: визначає асиметрію розподілу ймовірностей дійсної випадкової величини. Може мати додатне та від'ємне значення. Обчислюється за формулою:

$$s = \frac{E(x-\mu)^3}{\sigma^3} \quad (2.17)$$

9. Середньоквадратичне значення: міра середньої зміни конкретних значень від середнього їх значення. Обчислюється за формулою:

$$RMS = \sqrt{\frac{1}{N} \sum_{n=1}^N x_n^2} \quad (2.18)$$

10. Модель авторегресії: являє собою тип випадкового процесу і описує кожну вибірку сигналу як суму лінійних комбінацій попередніх вибірок і похибку білого шуму. Обчислюється за формулою:

$$x_n = -\sum_{i=1}^p a_i \cdot x_{n-i} + w_n \quad (2.19)$$

де  $x_n$  це вибірки сигналу,  $a_i$  - фактор авто регресії,  $w_n$ - білий шум,  $p$  - степінь авто регресії.

					ДП 6416.03.000 ПЗ	Арк.
						29
Змн.	Арк.	№ докум.	Підпис	Дата		

### 2.2.2.2. Особливості частотної області

Можна виділити такі особливості частотної області[20]:

1. Медіана частота: частота для якої спектр поділяється на частини з рівними потужностями.

Обчислюється як:

$$\sum_{j=1}^{FMD} p_j = \sum_{j=FMD}^m p_j = \frac{1}{1} \sum_{j=1}^M p_j \quad (2.20)$$

де  $p_j$  – потужність сигналу на частоті  $j$ .

2. Середня частота: зважена сума, розрахована як добуток частоти і потужності спектру і поділена на суму інтенсивності спектрограми, як у формулі:

$$FMN = \frac{\sum_{j=1}^M f_j \cdot p_j}{\sum_{j=1}^M p_j} \quad (2.21)$$

де  $f_j$  – частотний спектр на частоті  $j$ .

3. Загальна потужність: сума спектра потужностей, віднята від постійної величини нульового спектрального моменту

$$TP = \sum_{j=1}^K p_j - SMO \quad (2.22)$$

де  $p_j$  – потужність для частот  $j = 1..K$ , SMO – нульовий спектральний момент.

4. Середня потужність: середня потужність в спектрі потужності сигналу, розраховується за наступною формулою:

$$MNP = \frac{\sum_{j=1}^K p_j}{K} \quad (2.23)$$

5. Коефіцієнт спектра потужності: коефіцієнт потужності  $P_0$ , який є максимальним значенням спектра потужностей  $P$ , який є повною енергією спектра потужностей:

$$PSR = \frac{P_0}{P} = \frac{\sum_{f_0-n}^{f_0+n} p_j}{\sum_{-\infty}^{\infty} p_j} \quad (2.24)$$

### 2.2.3. Методи зменшення розмірності отриманих даних

Зменшення розмірності характеристик корисно з двох причин. По-перше, що характерно для деяких функцій, вони дають зайву інформацію. По-друге,

					ДП 6416.03.000 ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

якщо якісь особливості не мають значення, вони можуть негативно впливати на дискримінаційну здатність класифікатора. Таким чином, використання методики зменшення розмірності для передачі максимально чистої інформації до класифікатора покращить ефективність.

Методи зменшення розмірності можна розділити на два основних типа: на той, що фокусується на перетворення даних, відображаючи їх у меншій розмірності, та інший – що підбирає підмножини. Для аналізу обрані алгоритми аналіз основних компонентів (PCA) [21] та алгоритм RELIEF [22].

### 2.2.3.1 Метод головних компонент (PCA)

PCA зазвичай використовується для розпізнавання шаблонів для зменшення розмірності та створення функцій. Вихідний простір лінійно перетворюється на простір нижнього виміру шляхом проектування  $N$ -мірних даних на  $M$  власних векторів ( $M \leq N$ ), отриманих з матриці коваріації, що відповідає власним більшим значенням. Навіть у випадку, коли вибрані всі власні вектори  $i$ , таким чином, немає розмірності скорочення, можна досягти покращення ефективності класифікації у випадку, коли PCA породжує некорельовані функції [21].

PCA - це метод, який використовується для ідентифікації шаблонів даних, оскільки він перетворює дані, щоб зробити акцент на їх схожості та відмінності. Використання PCA стає ще важливішим у даних великих розмірностей, де графічне зображення неможливе. Більше того, PCA допомагає стиснути дані з так як виконується зменшення розмірності без втрати інформації [22].

Оригінальний простір даних перетворюється на новий, який обертається так, що його осі будуть вказувати у напрямку з найбільшою дисперсією даних. Ці осі називаються основними компонентами (ОК) та сортуються за

дисперсією, що означає, що перший головний компонент (ОК 1) представляє напрямок найвищої дисперсії даних. Більше того, ОК 2 представляє напрямок найвищого значення решти дисперсій, який є ортогональною для ОК 1. Це стосується всіх інших ОК, які можуть створити новий простір покриття бажаного рівня дисперсії. Слід зазначити, що кожен з

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		31



основних компонентів є лінійним поєднання всіх оригінальних змінних. Перехід з тривимірного простору (x, y, z) у двовимірний (ОК 1, ОК 2) можна зрозуміти за допомогою рисунка 2.9 [24].

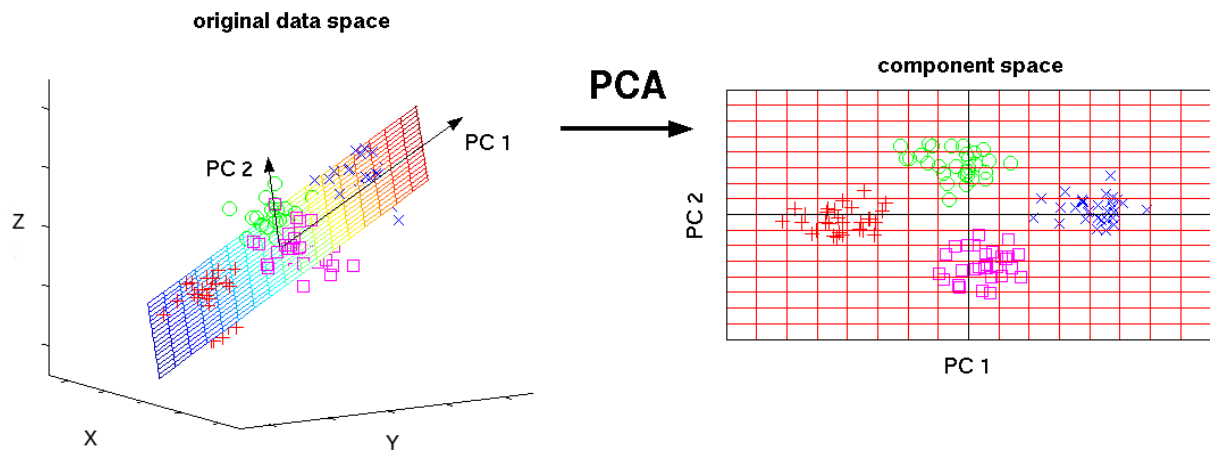


Рисунок 2.9 – перетворення PCA

#### Алгоритм PCA [23]

1. Отримайте характеристики у  $N$  вимірах та виконайте попередню обробку характеристики, віднімаючи середнє значення для кожного виміру (атрибута). Отже, якщо кожна функція має вектор  $x_i$  для  $i \in [1, N]$ , де всі значення мають середнє значення  $x_i$ . Це створює набір даних із нульовим середнім значенням. Отримана таблиця називається *DataAdjust*.
2. Розрахуйте матрицю  $N \times N$ .
3. Розрахуйте власні значення та власні вектори (матриця коваріації квадратна). Відсортуйте власні вектори від найвищої до найнижчої цінності, в порядку важливості для власних значень.
4. Отримуйте максимальний власний вектор, який є головним компонентом (ОК). Ігноруйте тих, хто має найменшу значимість, особливо якщо вони не мають важливої інформації. Тому, якщо є  $N$  виміри власних значень та власних векторів виберіть перший  $P$  власне значення, він буде включати в набір даних лише з  $P$  вимірами.
5. Побудуйте векторну ознаку:  $FeatureVector = (eig_1 eig_2 \dots eig_p)$
6. Отримайте новий набір даних. Помножте матрицю транспонування *FeatureVector* на обернену матрицю початкового набору даних:

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		32

$$FinalData = FeatureVector^T \times DataAdjust^T$$

### 2.2.3.2. Алгоритм RELIEF

RELIEF вважається одним із найуспішніших алгоритмів завдяки простому та ефективному оцінюванню якості функцій. [22].

Алгоритм RELIEF досліджує функції, що мають статистичне значення для цільової концепції з використанням навчальних даних  $D$ , розмір вибірки  $m$  та порігу  $\tau$  ( $\tau \in [0,1]$ ). Якщо два екземпляри різні, це може бути виражено функцією  $diff$ , яка використовує два екземпляри  $x, y$  як вхідні дані.

Функція  $diff$  у числовому вигляді (ціле чи дійсне):

$$diff(x_k, y_k) = \frac{(x_k - y_k)}{no_k} \quad (2.25)$$

де  $no_k$  - число, яке використовується для нормалізації значень, які відрізняються і знаходяться в межах  $[0, 1]$ .

RELIEF вибирає зразок у  $m$  трійках ( $X$ , Near-Lose, Near-hit). Для того, щоб вибрати останні два, у деяких випадках для оновлення ваги використовується евклідова відстань. Вибрані функції - це ті, які є набрали вище, ніж визначений користувачем поріг.

Алгоритм RELIEF [25]

1. Розділіть  $D$  на додатні ( $D^+$ ) і від'ємні ( $D^-$ ) екземпляри,  $W = (0, 0, \dots, 0)$

2. Для  $i = 1$  до  $m$

Оберіть випадковий екземпляр  $X \in D$

Виберіть випадковим чином  $Z^+ \in D^+$  найближчий до  $X$

Виберіть випадковим чином  $Z^- \in D^-$  - найближчий до  $X$

якщо ( $X$  - додатній екземпляр)

тоді  $Near - hit = Z^+$ ;  $Near - miss = Z^-$

інакше  $Near - hit = Z^-$ ;  $Near - miss = Z^+$

для  $i = 1$  до  $p$  оновити ваги

$$w_i = w_i - diff(x_i, Near - hit)^2 + diff(x_i, Near - miss)^2$$

3.  $Relevance = \left(\frac{1}{m}\right) W$

для  $i = 1$  до  $p$

					ДП 6416.03.000 ПЗ	Арк.
						33
Змн.	Арк.	№ докум.	Підпис	Дата		

якщо ( $relevance_i \geq \tau$ )

тоді  $f_i$  є актуальною особливістю

інше  $f_i$  є неактуальною особливістю

RELIEF, як визначено вище, вибирає зразок з  $m$  трійки екземпляра  $x$ , його екземпляра Near-hit та  $a$  екземпляра Near-miss. RELIEF використовує  $p$ -мірну евклідову відстань для вибору Near-hit та Near-miss. Він оновлює ваговий вектор характеристик  $W[10]$  для кожного триплета  $m$  і обчислює середнє значення вагової векторної актуальності). Врешті-решт, алгоритм вибирає характеристики із середньою вагою («рівень відповідності») вище встановленого порогу  $\tau[10]$ . Вищезгаданий алгоритм націлений на проблеми бінарної класифікації [25].

					ДП 6416.03.000 ПЗ	Арк.
						34
Змн.	Арк.	№ докум.	Підпис	Дата		

## ВИСНОВКИ ДО РОЗДІЛУ 2

У даному розділі були розглянуті сигнали, їх класифікація за фізичною природою та залежно від функції. Було розглянуто спосіб перетворення сигналу з аналогового у дискретний, описано принцип та алгоритм роботи АЦП. Визначено етапи перетворення: дискретизація, квантування та кодування.

Крім цього, було розглянуто методи аналізу отриманих сигналів, а саме метод ЕМД, Описані особливості отриманих сигналів.

Також були визначені методи зменшення розмірності отриманих даних. Аналіз показав, що одними з самих поширених методів є метод PCA та RELIEF. Були описані алгоритми цих методів.

					ДП 6416.03.000 ПЗ	Арк.
						35
Змн.	Арк.	№ докум.	Підпис	Дата		

## РОЗДІЛ 3. ПРОЕКТУВАННЯ

### 3.1 Механічна частина

Першим питанням при розробці манупілятора був вибір механічної частини проекту, тобто самої руки. Було вирішено друкувати руку на 3D принтері. Сьогодні 3D друк стає все більш поширеним у багатьох сферах діяльності. Технологія дозволяє створювати потрібні об'єкти досить швидко і дешево, маючи при цьому деякі переваги перед традиційними методами виробництва, такими як лиття, фрезерівка, штамповка та інші. Основними перевагами є:

- Швидкість виробництва
- Вартість виробництва
- Якість отриманої продукції
- Висока точність
- Доступність
- Простота

Так як для створення моделі був потрібен одичний екземпляр, швидкість його друку на 3D принтері буде у багато разів перевершувати строки створення подібного проекту за допомогою інших методів виробництва. Наприклад, за час фрезування однієї деталі, більша кількість 3D принтерів встигне надрукувати декілька таких самих деталей.

Аналогічна ситуація і з вартістю друку. Для виробництва декількох деталей знадобиться не дуже багато пластику для принтеру, що буде значно дешевше ніж матеріали та обладнання для інших видів виробництва.

Залежно від моделі принтера та налаштувань друку, можна досягти досить великої точності надрукованих деталей. Також можливо виробляти складні деталі, які неможливо зробити використовуючи інші методи. Перевагою друку є можливість створювати рухомі елементи та складні деталі з прихованими порожнинами та складною структурою заповнення деталі.

					ДП 6416.03.000 ПЗ	Арк.
						36
Змн.	Арк.	№ докум.	Підпис	Дата		

Недоліками друку є менша точність дотримання заданих розмірів та обмеженість габаритів деталі в залежності від самого принтера.

Ці недоліки є несуттєвими для даного проекту, тому 3D печать обрана як найкращий варіант реалізації механічної частини руки.

Як оптимальний варіант було обрано французський open-source проект робота inMoov, а саме окремо руку. Всі деталі надруковані на 3D принтері з пластика CoPET. На рисунку зображені надруковані деталі цієї моделі, які потім потрібно зібрати:



Рисунок 3.1 - деталі руки-маніпулятора

## 3.2 Вибір апаратного забезпечення

Існує велика кількість мікроконтролерів і мікропроцесорних пристроїв, на яких можна реалізувати проект руки-маніпулятора. Більшість з пристроїв пропонує схожий функціонал і дає простий інтерфейс для програмування.

Однією з найпоширеніших платформ є Arduino. Ця платформа має декілька важливих переваг:

- Низька вартість. Порівняно з іншими варіантами, Arduino має відносно невелику вартість. Також є копії плат, які коштують ще дешевше.
- Кросплатформеність. Програмне забезпечення Arduino працює на всіх ОС.
- Зручне середовище програмування.
- Програмне забезпечення з відкритим кодом. Є можливість використовувати різні бібліотеки і використовувати вставки мови C.
- Розширюване відкрите апаратне забезпечення. Платформа побудована на базі мікроконтролера Atmel Atmega.
- Велика кількість деталей та розширень.
- Різні версії платформи з різною продуктивністю і ціною.

### 3.2.1 Огляд платформи Arduino

Arduino – це апаратно-обчислювальна платформа. Її основними компонентами є плата вводу/виводу і середовище розробки Processing/Wiring. Arduino може використовуватись як для розробки автономних об'єктів, так і для взаємодії з програмним забезпеченням комп'ютера. Плати побудовані з мікроконтролера Atmel (ATMega) та елементів обв'язки для взаємодії з іншими схемами. Крім цього, на платах є стабілізатор напруги на 5 або 3,3 В. Кварцевий резонатор (в деяких версіях використовується керамічний резонатор) здійснює тактування на частоті 8 або 16 МГц. Також у мікроконтролер вшитий завантажувач BootLoader.

Всі плати Arduino програмуються через інтерфейс RS-232. Цей інтерфейс також широко відомий як послідовний, або COM-порт. Для конвертації сигналів RS-232 у TTL (транзисторно-транзисторна логіка) та навпаки, на платі є інвертуюча схема. Наприклад, у версії Arduino Uno для конвертування

					ДП 6416.03.000 ПЗ	Арк.
						38
Змн.	Арк.	№ докум.	Підпис	Дата		

використовується мікроконтролер ATmega 8. Завдяки цьому платформа може визначатись при підключенні до ПК як периферійні девайси, наприклад клавіатура чи джойстик. Плати Arduino дозволяють використовувати більшу частину входів та виходів мікроконтролера для взаємодії з зовнішніми схемами. Наприклад на платі Uno є 13 цифрових входів виходів і 5 аналогових входів. Сигнали доступні через контактні площадки або штирьові роз'єми на платі. Також для розширення доступні різні зовнішні плати, так звані шилди, які приєднуються до плати через штирьові роз'єми. Максимальний струм, який може бути поданий або отриманий – 40 мА. Кожен вхід або вихід містить всередині резистор на 20-50 кОм, який за замовчуванням не підключений.

Окрім загальних операцій, деякі з входів та виходів здатні виконувати спеціальні операції:

- Зовнішні переривання: контакти 2 та 3 використовуються для генерації сигналу переривання, синхронізації або зміни значення.
- Серійні: контакти 0 (RX) та 1 (TX) використовуються для отримання та передачі послідовностей TTL.
- SPI: контакти 10 (SS), 11 (MOSI), 12 (MISO) и 13 (SCK) налаштовані для зв'язку SPI з використанням SPI бібліотек. SPI в Arduino – це один з основних протоколів обміну даними між платою Arduino та під'єднаними пристроями. Цей протокол використовується для багатьох периферійних пристроїв.
- ШІМ: контакти 3, 5, 6, 9, 11 забезпечують 8-бітний ШІМ вихід. ШІМ (широтно-імпульсна модуляція).
- Світлодіод: до контакту 13 підключений світлодіод, що показує наявність контакту.

Крім того, є 6 аналогових входів (A0 - A5) з 10-бітовою роздільною здатністю.

Крім того, наступні контакти мають особливий функціонал:

- AREF: це опорна напруга для аналогових входів.
- TWI (I2C): штифт A4 або SDA та штифт A5 або SCL. За допомогою шини I2C можливо під'єднати на 2 проводи велику кількість різних периферійних

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		39



пристроїв. При цьому SDA відповідає за передачу інформації, а SCL – за тактування шини.

- Скидання: скидає мікроконтролер.

На рисунку 3.2. зображена схема підключень Arduino та ATmega328.

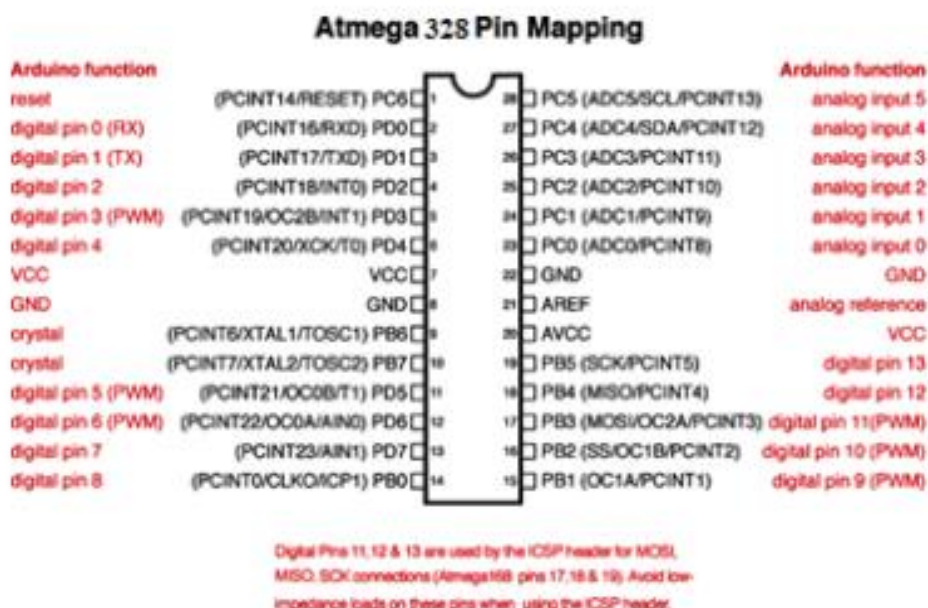


Рисунок 3.2 - відображення контактів Arduino

### 3.2.1.1. Живлення

Плата може живитися як через зовнішній блок живлення, так і через USB-з'єднання, що є обраним за замовчуванням. Блок живлення може бути як акумулятором, так і адаптером змінного струму. Arduino оснащена роз'ємом живлення 2,1 мм. Також можна під'єднати провід акумулятора до штифтів плати, VIN та GND від роз'єму POWER.

Зовнішнє джерело живлення може змінюватись від 6-20 В. Відповідна область роботи - від 7 до 12 В [26]. Це є через дві причини. У випадку, коли  $V_{dd} < 7V$ , 5-контактний штир подаватиме менше напруги, що призводить до нестабільності плати. У випадку, коли  $V_{dd} > 12V$ , є ймовірність перегріву регулятора напруги, що може призводити до пошкодження плати.

На платі є наступні силові штифти:

- VIN: контакт працює як напруга на вході живлення для плати із зовнішнього джерела. Його можна використовувати або для живлення, або для зчитування напруги, що подається через роз'єм живлення.
- 5V: забезпечує вихід 5В, який в свою чергу забезпечує регулятор. Потужність, що подається на гніздо постійного струму (Напруга постійного струму) і контактний VIN може варіюватися від 7 до 12В і 5В отриманих роз'ємом USB.
- 3V3: Цей контакт забезпечує регульовану напругу 3,3 В з максимальним струмом 50 мА струму.
- GND: заземлювач.
- IOREF: Цей штифт забезпечує інформацію про напругу, на основі якого заснована робота мікроконтролера. Цей штифт зчитується платою розширення, який вибирає відповідний рівень потужності або дозволяє використовувати перетворювач рівнів, що дозволяє працювати з напругою 3,3 В або 5 В.

### 3.2.1.2. Зв'язок

Є багато способів зв'язку між Arduino з іншим Arduino або взагалі зв'язку мікроконтролера з комп'ютером. Послідовний зв'язок UART TTL (5V) - один із способів забезпечувати зв'язок через цифрові виводи 0 (RX) та 1 (TX). Послідовний зв'язок через USB здійснюється через ATmega16U2, який відображається до комп'ютера як віртуальний порт зв'язку. Стандартні USB-драйвери COM без потреби використання будь-яких зовнішніх, використовуються для вбудованого програмного забезпечення 16U2. Також доступний послідовний монітор для надсилання та отримання текстових даних через Arduino. На платі є два світлодіоди, RX і TX, що допомагають візуальному контролю передачі даних через USB-з'єднання до комп'ютер і USB-серійний чіп [26].

Для проекту було обрано платформи Arduino Nano та Uno (Atmega328).

					ДП 6416.03.000 ПЗ	Арк.
						41
Змн.	Арк.	№ докум.	Підпис	Дата		

### 3.2.2. Огляд обраних деталей

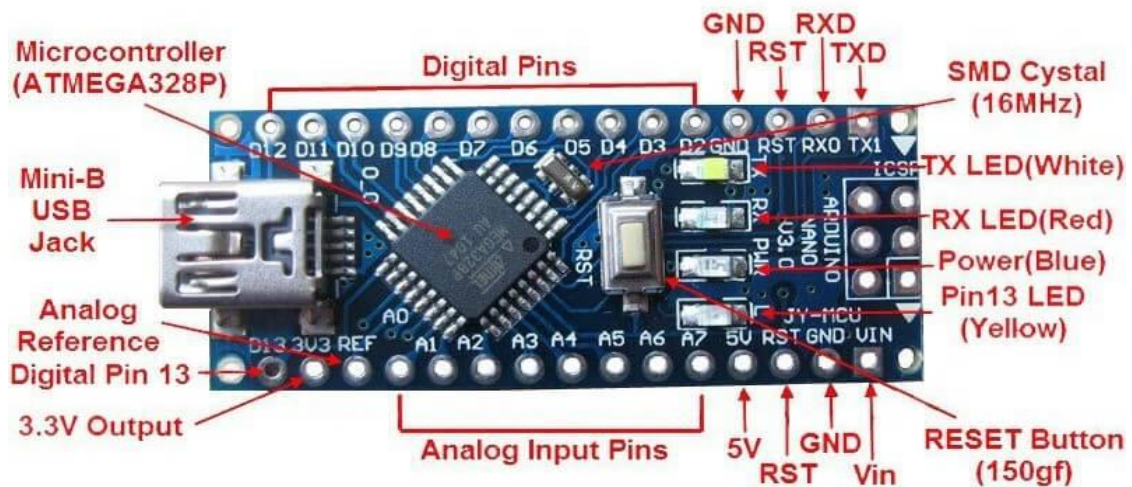


Рисунок 3.3 - схема Arduino Nano

Характеристики:

Мікроконтролер — ATmega328P

Робоча напруга — 5 В

Пам'ять — 32 кб

Розміри — 45x18 мм

Цифрові входи\виходи – 14 (6 з них можуть використовуватись як ШИМ-виходи)

Тактова частота – 16 МГц

Вага — 7 г

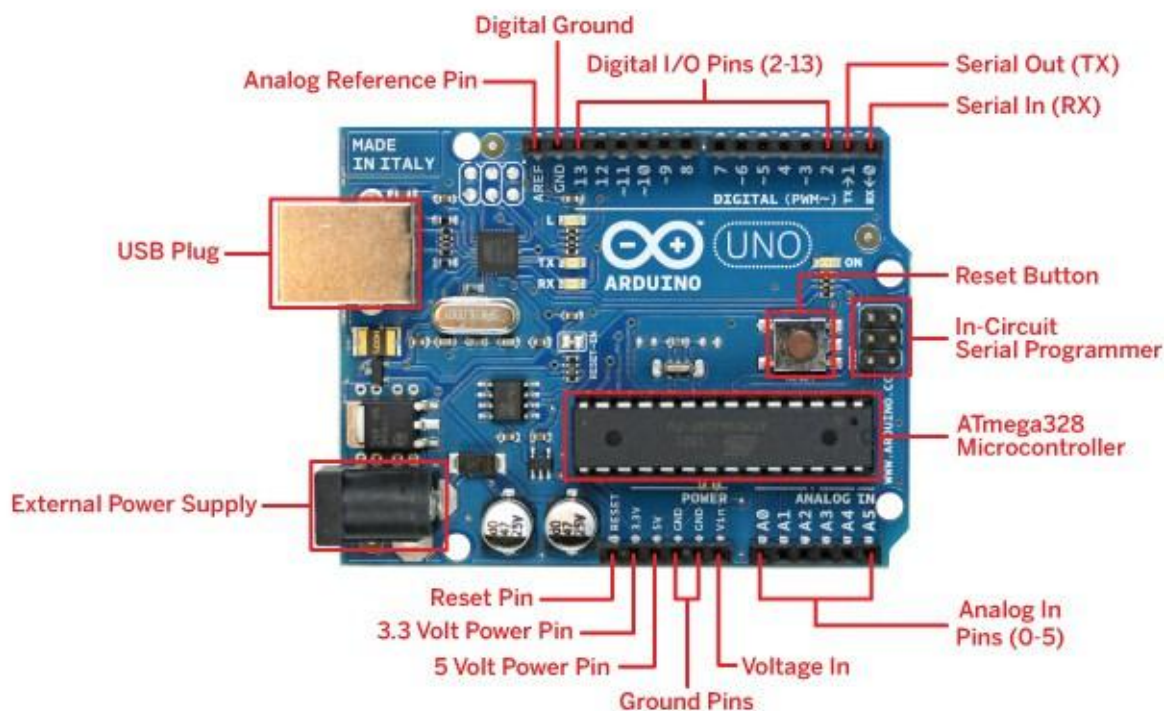


Рисунок 3.4 - схема Arduino Uno

Структурна схема Arduino Uno наведена у додатку В.

Характеристики:

Мікроконтролер — ATmega328

Робоча напруга — 5 В

Пам'ять — 32 кб

Розміри — 68.6x53.4 мм

Вага — 25 г

Для керування механічною частиною були обрані сервоприводи. Вони дозволять точно управляти рухами. Принцип роботи сервопривода такий: [28]

- Сервопривід отримує імпульсний сигнал - керуюче значення, яке визначає кут повороту качалки сервопривода,
- Блок управління починає порівняння отримані параметри зі значенням на своєму датчику,
- Залежно від результату порівняння БО повертає сигнал, який зумовлює, яку дію необхідно виконати: повернути, прискоритися або сповільнитися, щоб порівнювані показники стали однаковими.

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		43

Більшість сучасних сервоприводів побудовані за одним принципом і складаються з таких складових частин: вихідний вал, шестерні редуктора, двигун постійного струму, потенціометр, друкована плата і керуюча електроніка. На рисунку 3.5 зображено внутрішню будову сервопривода.

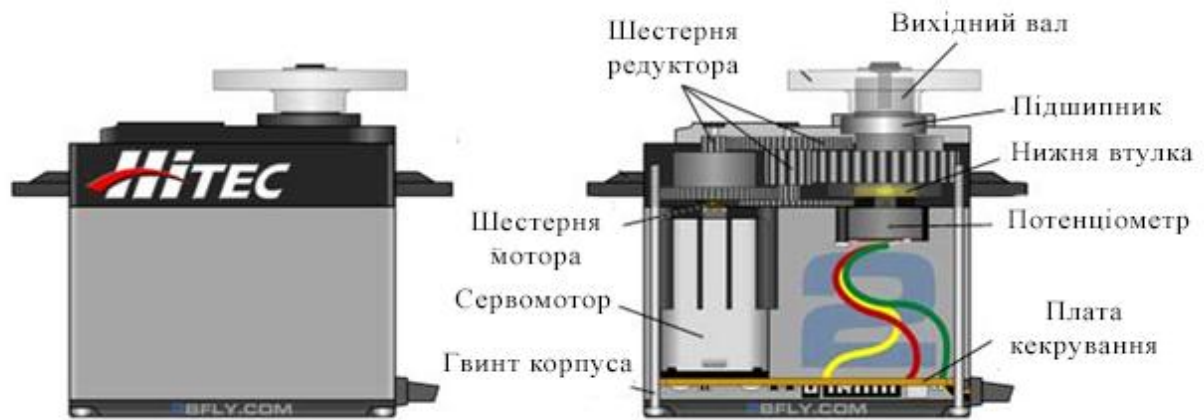


Рисунок 3.4 - будова сервопривода

Редуктор разом з мотором утворюють привід. Щоб трансформувати напругу в механічний поворот, потрібен електродвигун. Редуктор - конструкція з шестерень - перетворює крутний момент і служить для зниження швидкості обертання двигуна, так як часто вона настільки велика, що зовсім не годиться для практичного застосування. Разом з вмиканням і вимиканням електродвигуна обертається і вихідний вал, до якого закріплюється качалка - її, в свою чергу, кріплять до потрібного рухомого елемента. Саме качалка буде задавати рух моделі, а для цього в пристрої сервоприводу передбачений потенціометр - датчик, здатний перетворити кут повороту назад в електро-сигнал.

Однак, одним з головних елементів є плата управління, яка являє собою електронну схему. Саме вона отримує електричний імпульс, аналізує отриманий сигнал з даними потенціометра і вмикає та вимикає електродвигун.

Ось як влаштований сервопривід і робота його елементів. До речі, в якості мотора в пристрої сервоприводу можуть використовуватися колекторні, колекторні Coreless і безколекторні двигуни.

Сервопривід отримує імпульсні сигнали, які проходять за спеціальним провідом від приймача. Частота таких сигналів становить 20мс, а їх тривалість може варіюватися в межах 0,8-2,2мс. Щоб у зрозуміти як сигнал трансформується в переміщення качалки, потрібно проаналізувати стандартну схему, зображену на рисунку 3.5.

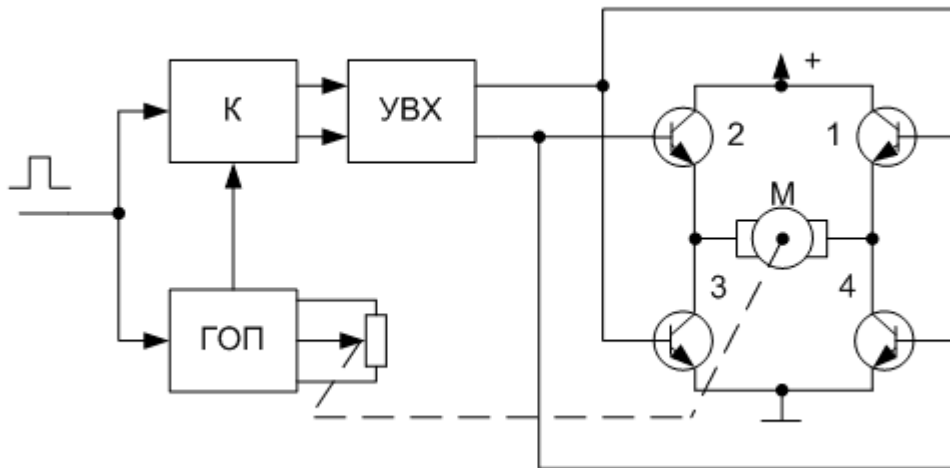


Рисунок 3.5 - схема сервопривода

ГОП - генератор імпульсу (до нього приєднаний потенціометр), К - компаратор, ПВЗ - пристрій вибірки-зберігання, М - електричний мотор, який охоплюється діагоналлю силового мосту.

Розберемо більш детально, як працює сервопривід. Отже, імпульсний сигнал надходить від ресивера на компаратор і в той же час активує ГОП. Тривалість опорного імпульсу залежить від розташування об'єкта потенціометра, який з'єднаний з вихідним валом фізично. Коли качалка знаходиться в середній позиції, довжина сигналу становить 1,5мс, якщо ж положення крайнє - 0,8 або 2,2 мс.

Керуючий сигнал і опорний імпульс аналізуються компаратором, який розраховує їх різницю величин (розрахунок ведеться по тривалості імпульсів). Саме довжина різності імпульсу і визначає наскільки «очікуваний» і «фактичний» стан керма збігається. Отриманий показник зберігається в якості потенціалу в ПВЗ.



Для проекту знадобляться 5 сервоприводів. Обрано модель MG996R 15 кг, яка зображена на рисунку 3.6.. Всі внутрішні частини цього сервоприводу зроблені з металу, що робить його надійним. Важливо надати стабільне живлення від окремого джерела живлення.



Рисунок 3.6 - MG996R

Характеристики:

Робоча напруга — 4.8-7.2 В

Кут нахилу — 120

Розміри — 40x19x43 мм

Вага — 55 г

Для керування сервоприводами потрібен модуль PCA9685 — 16 канальний контролер з налаштуванням частоти ШИМ, який зображено на рисунку 3.7. За допомогою ШИМ контролера можна керувати кожний сервоприводом. Для керування використовується шина I2C. До плати можна підключили 16 сервоприводів.

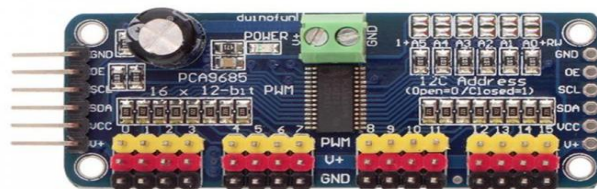


Рисунок 3.7 - PCA9685

Керувати сервомоторами з бібліотекою Servo Arduino досить легко, але кожен витрачає дорогоцінний штифт - не кажучи вже про деяку потужність обробки Arduino. 16-канальний 12-бітний ШИМ / сервопривід Adafruit може керувати до 16 сервоприводами через I2C лише з 2 штирями. Бортовий ШИМ-

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		46

контролер буде керувати всіма 16 каналами одночасно без додаткових накладних обробок Arduino. Більше того, можна скласти ланцюжок з до 62 з них, щоб керувати до 992 сервоприводів - все з тими ж 2-ма штирями.

ШІМ (широтно-імпульсна модуліція) – це спосіб управління поданням потужності до нагрівки. Управління засновано на зміні тривалості імпульса при незмінній частоті проходження імпульсів.

ШІМ контролер – це пристрій, що керує силовими ключами. При цьому керування відбувається за допомогою інформації, яка отримана по ланцюгах зворотного зв'язку по струму або напрузі, це потрібно для стабілізації вихідних параметрів. Отже, ШІМ контролер – це пристрій, що керує ШІМ сигналом.

Основні характеристики такого сигналу це частота імпульсу, що залежить від перетворювача, так коефіцієнт заповнення – частина часу від періода, коли на виході контролера формується сигнал, який менше 1.

Основними складовими ШІМ контролера є генератор імпульсів, джерело опорної напруги, компаратор, транзистори.

На прикладі обраного контролера, розглянемо його будову більш детально. З двох боків є два набори керуючих вхідних штифтів. Обидві сторони однакові. Можна використовувати будь-яку сторону, яка буде більш зручною, також можна легко зв'язати ланцюги, з'єднавши дві бічні сторони

#### Штифти живлення

GND - Це вивід живлення та сигналу заземлення, який потрібно підключити.

VCC - це логічний штифт живлення, потрібно підключити його до логічного рівня, який ви хочете використовувати для виходу PCA9685, може бути під'єднано 3 - 5 В максимально.

V + - Це додатковий штифт живлення, який подаватиме розподілену потужність на сервоприводи. Якщо додаткове живлення для сервоприводів не використовується, він може залишатись не під'єднаним. У такому разі цей штифт взагалі не використовуватиметься чіпом. Також можна подати живлення до 2-контактного клемного блоку у верхній частині плати. Слід надати 5-6 В постійного струму, якщо використовуються сервоприводи.

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		47



## Контрольні штифти

SCL - тактовий штифт I2C, підключіть до мікроконтролерів тактову лінію I2C. Може використовувати логіку 3V або 5V, і має слабку підтягування до VCC

SDA - контактний контакт I2C, підключається до лінії даних I2C мікроконтролера. Може використовувати логіку 3V або 5V.

OE - вихідний режим. Можна використовувати для швидкого відключення всіх виходів. За замовчуванням це необов'язковий штифт.

## Вихідні порти

На платі є 16 вихідних портів. Кожен порт має 3 штифта: V +, GND та вихід ШІМ. Кожна ШІМ працює повністю незалежно, але всі вони повинні мати однакову частоту ШІМ. Тобто для світло діодів використовується 1 КГц, а сервоприводам потрібно 60 Гц, тому не можна використовувати половину портів для світлодіодів при 1 КГц і половину для сервоприводів при 60 Гц. Максимальний струм на один контакт 25 мА. На платі є резистори 220 Ом з усіма ШІМ-штифтами, і логіка виходу така ж, як VCC.

Для зняття електроміограми використовується пристрій Myoware muscle sensor , зображений на рисунку 3.8 і електроди для ЕМГ.

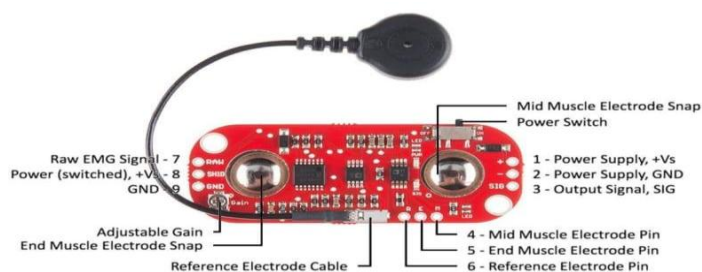


Рисунок 3.8 - схема Myoware muscle sensor

Характеристики пристрою подані у таблиці 3.1. [29]

Таблиця 3.1 Характеристики Myoware muscle sensor

Параметр	Min	Typical	Max
Напруга живлення	2.9В	3.3-5В	5.7В
Регульований коефіцієнт посилення потенціометра	0.01Ом	50 кОм	100 кОм

Продовження Таблиця 3.1 Характеристики Myoware muscle sensor

Параметр	Min	Typical	Max
Вихідна напруга сигналу	0	-	0
Вхідний опір	-	110 ГОм	-
Струм живлення	-	9мА	14 мА
Загальний коефіцієнт відхилення	-	110	-
Зміщення вхідних даних	-	1 pA	-

При підключенні цього датчика до Arduino, можна використовувати serial plotter для відображення ЕМГ сигналів. Отриманий таким чином графік зображено на рисунку 3.9.

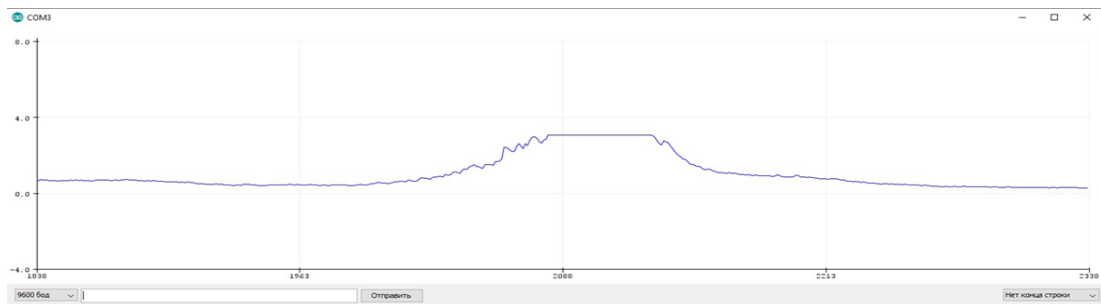


Рисунок 3.9 - відображення ЕМГ сигналів

Для бездротової передачі даних, знятих за допомогою датчика, буде використовуватись радіо модуль nRF24L01, який зображено на рисунку 3.10.

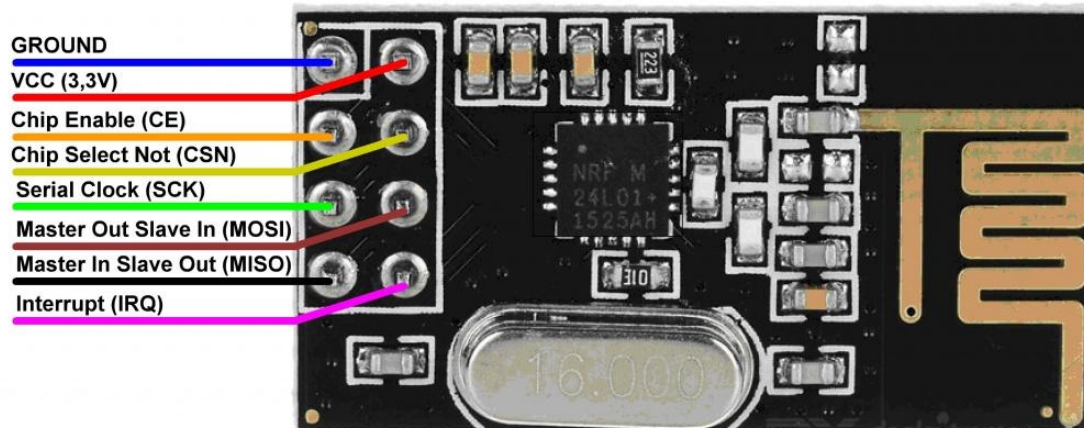


Рисунок 3.10 - модуль nRF24L01

Цей модуль призначений для прийому і передачі даних по радіоканалу, на дозволеному діапазоні частот. Цей діапазон розбитий на 128 каналів з кроком 1 МГц: від 2400 ГГц до 2527 ГГц. Модуль дозволяє обрати будь який канал прийому або передачі даних. На кожному каналі можна сформувати мережу з 1 передатчика та 6 прийомників. Модуль підключається до шини SPI. Модулю програмно задається роль передавача або приймача, але в ході виконання програми, цю роль можна міняти. Є можливість задавати: потужність передачі (-18 дБм, -12 дБм, -6 дБм, 0 дБм), швидкість передачі (250 кб / с, 1 Мб / с, 2 Мб / с), номер каналу (0-127), ідентифікаційні номери (5 байт) і т.д. Модуль не вимагає підключення антени, тому що вона вбудована присутній на ПП модуля.

Споживані модулем струми не перевищують 13 мА (як при передачі, так і при прийомі). Достовірність отриманих даних забезпечується передачею циклічно надлишкового коду CRC Реалізовано функціонал контролю доставки даних. Приймач, після успішного отримання пакету даних, відправляє передавача пакет підтвердження прийому.

### 3.3. Вибір програмного забезпечення

Для програмування буде використовуватись Arduino IDE. Це крос-платформний додаток, який записується у функції C та C++. Додаток використовується для загрузки програм на платформу Arduino. IDE підтримує мови програмування C та C++. Проекти, що розробляються в Arduino IDE називаються скетчами.

Бібліотеки, що будуть використовуватись:

SPI:

Послідовний периферійний інтерфейс (SPI) – це протокол синхронної послідовної передачі інформації, який використовується мікроконтролерами для швидкого зв'язку з периферійними пристроями або іншими мікроконтролерами. При використанні цього інтерфейсу завжди є один головний пристрій, який керує іншими. Інтерфейс має 3 лінії, загальних для всіх пристроїв:

- MISO (Master In Slave Out) – підлегла лінія для передачі даних на керуючий пристрій.
- MOSI (Master Out Slave In) – лінія керуючого пристрою для пересилки даних на периферійні пристрої.
- SCK (Serial Clock) – тактові імпульси, що синхронізують передачу даних, які генеруються керуючим пристроєм.

Також є одна лінія, що є специфічною для кожного пристрою:

- SS (Slave Select) – контакт на кожному пристрої, який може використовуватись для керуючого пристрою для вмикання і вимикання певних пристроїв.

Існують 4 режими передачі даних. Ці режими контролюють зсув на вході і виході тактового сигналу даних і перевіряють, чи знаходяться таймер в режимі очікування. [27]

					ДП 6416.03.000 ПЗ	Арк.
						51
Змн.	Арк.	№ докум.	Підпис	Дата		

## ВИСНОВКИ ДО РОЗДІЛУ 3

У цьому розділі було обрано реалізація механічної частини проекту, найбільш вдалим варіантом став друк на 3D принетрі.

Було обрано апаратне забезпечення. Серед розглянутих платформ було обратно Arduino, так як вона є досить поширеною, зручною в користуванні і має безліч плат розширення та компонентів.

Для управління механічною частиною взяті сервоприводи MG996R, які є надійними, недорогими, та дозволяють точно контролювати рухи. Для керування всіма сервоприводами знадобиться ШІМ контролер PCA9685.

Для зняття ЕМГ сигналів буде використовуватись Myoware muscle sensor та електроди, що приєднуються до нього.

Для передачі даних від датчика на Arduino буде використовуватись модуль бездротового зв'язку nRF24L01.

Для програмування обрано Arduino IDE, крос-платформений додаток, що є зручним та функціональним.

					ДП 6416.03.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		52

## РОЗДІЛ 4. РОЗРОБКА ТА ТЕСТУВАННЯ

Для управління маніпулятором за допомогою ЕМГ сигналів побудовано систему, що складається з 2 підсистем:

Головний пристрій, що складається з платформи Arduino Uno, модуля з севоприводами та передатчика NRF24. Ця частина обробляє сигнали і виконує рухи. Схема подана на рисунку 4.1.

Пристрій на основі Arduino Nano, передатчика NRF24 та датчика Myoware muscle sensor, що зчитує та пересилає ЕМГ сигнали. Схема подана на рисунку 4.2. Схеми подані у додатках А та Б.

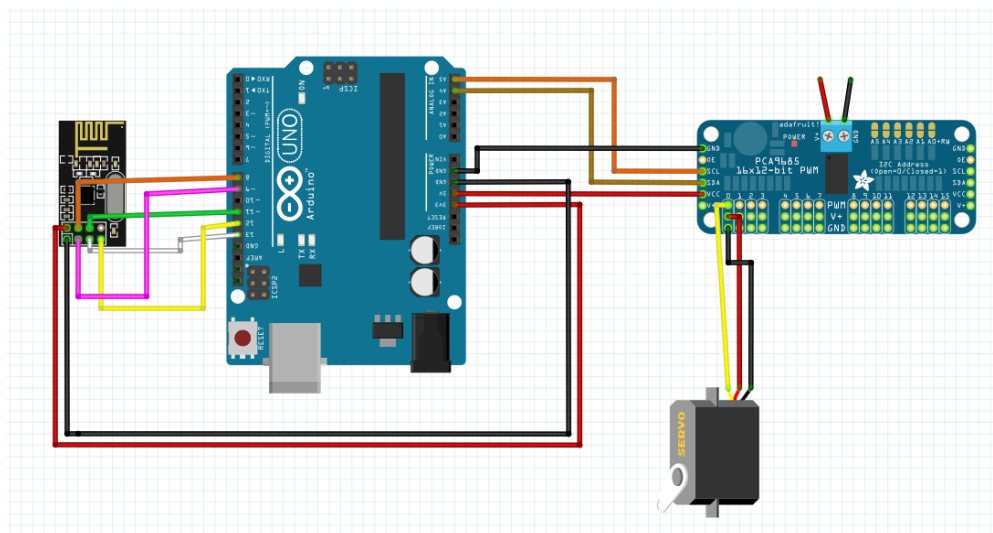


Рисунок 4.1 - схема головного пристрою

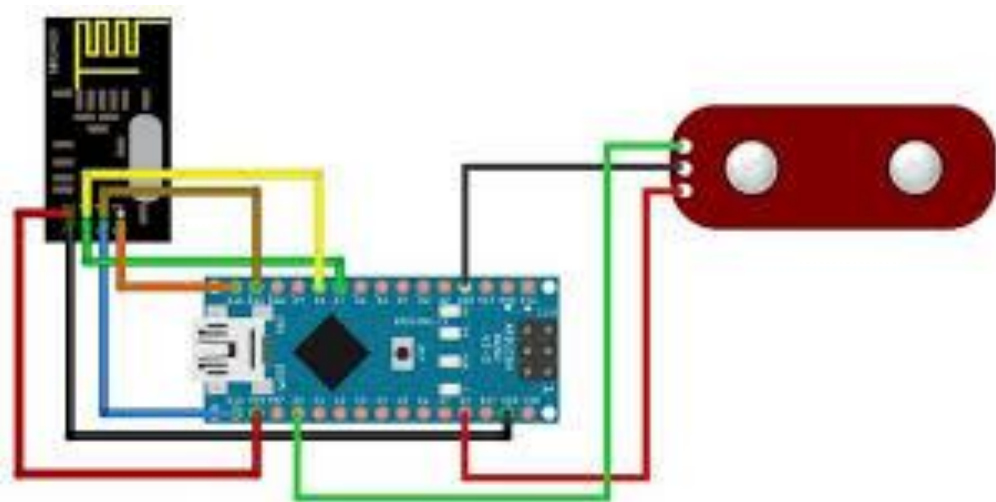


Рисунок 4.2 - схема підлеглого пристрою

Механічна конструкція повинна бути максиально простою та достатньо надійною для використання в повсякденній діяльності.

Принцип керування схожий на механізм множинної передачі, який зображений на рисунку 4.3. Він складається з кабелів, щоб проходили через пальці (шків) незалежно одне від одного, що дозволяє переміщати їх окремо один від одного.

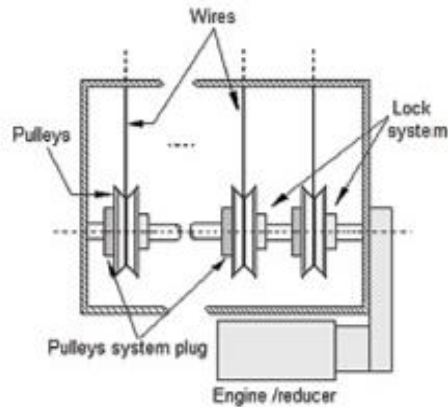


Рисунок 4.3 - механізм передачі рухів

На рисунку 4.4, 4.5 показана реалізація цього механізму для моделі руки. Як видно, через руку, від сервоприводів до пальців протягнуто волосінь. При зміні положення, вона натягується та переміщує пальці, в яких вона протягнена.



Рисунок 4.4 - механізм передачі рухів (1)



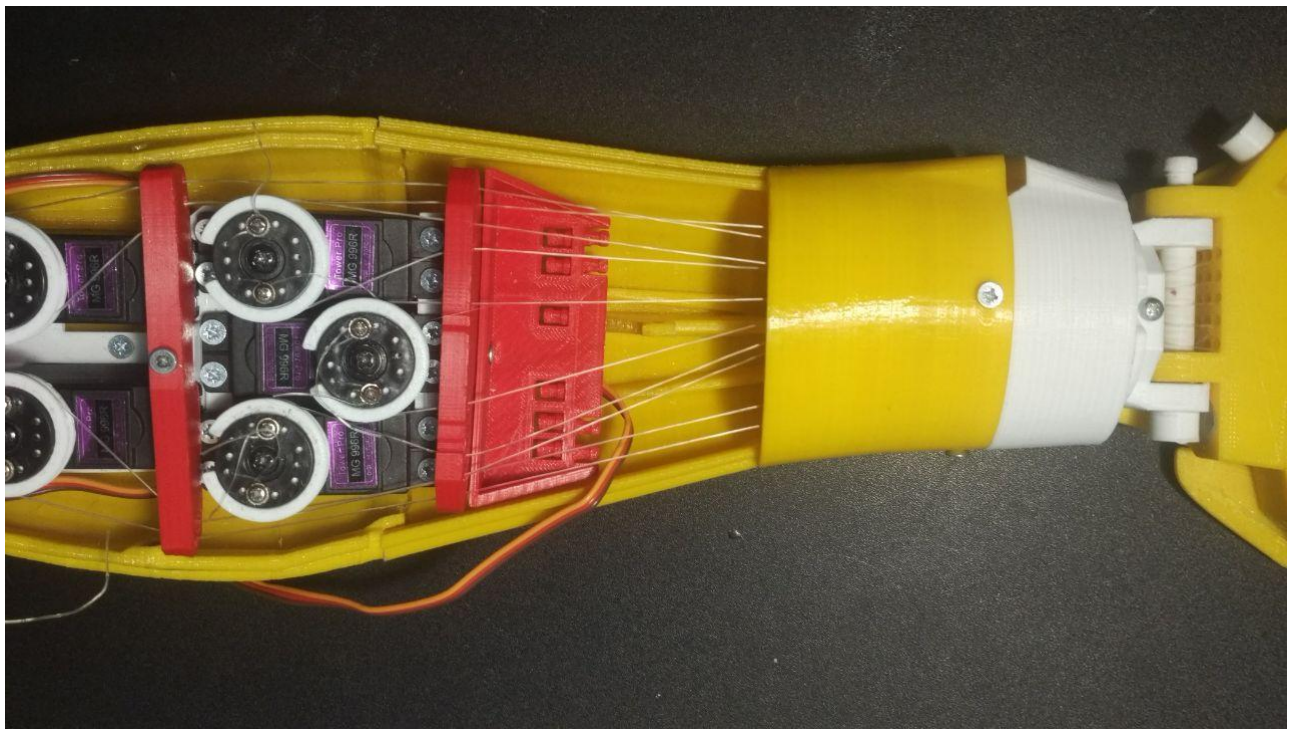


Рисунок 4.5 - механізм передачі рухів (2)

Для того щоб отримати правильні результати, важливо розмістити датчики, що реєструють м'язеву активність в правильних місцях. Існують багато варіантів розміщення, в залежності від потрібних м'язів та індивідуальних особливостей організму. На рисунку 4.6 видно один із варіантів розташування датчиків.

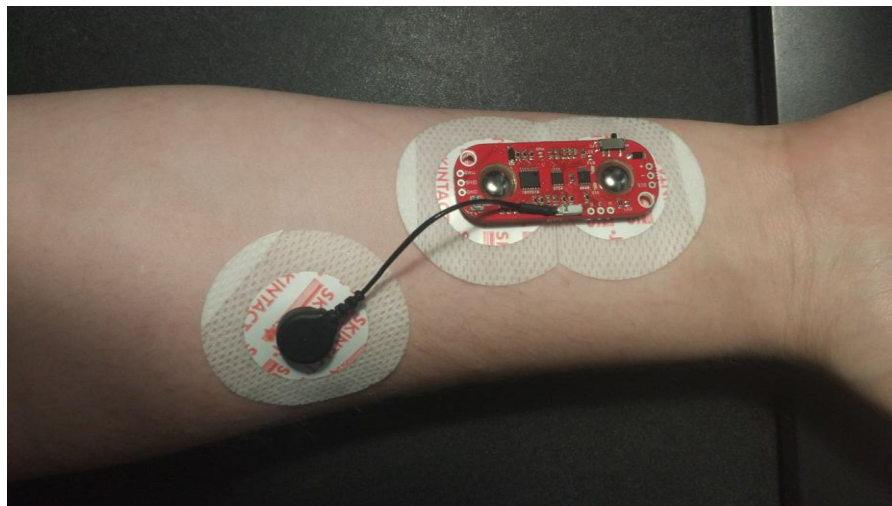


Рисунок 4.6 - розташування ЕМГ датчиків

Для тестування ЕМГ датчик можна підключити до Arduino. Для отримання сигналів, розроблено тестовий скетч.



```

void setup (){
    Serial.begin(9600);
}

}

void loop(){
    Serial.println(analogRead(0));
    delay(1);
}

```

Використовуючи плоттер Arduino, можна виводити отримані сигнали. Як видно на рисунку 4.5, при скороченні м'язів графік зростає.

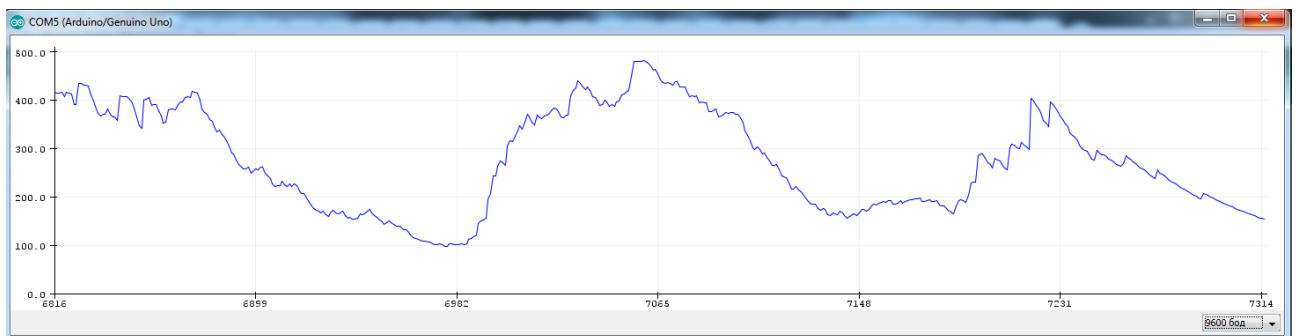


Рисунок 4.5 - вивід отриманих сигналів.

Далі було проведено тестування підключення сервоприводів до ШІМ контролеру. Написано скетч, що одночасно ставить всі сервоприводи в положення 0, а через 2 секунди змінює положення на 80 градусів. Використана бібліотека `Adafruit_PWMServoDriver`, для управління контролером. Після перевірки видно, що всі сервоприводи працюють коректно.

```

#include <Wire.h>

#include <Adafruit_PWMServoDriver.h>

Adafruit_PWMServoDriver pwm = Adafruit_PWMServoDriver();

#define MIN_PULSE_WIDTH 600#define MAX_PULSE_WIDTH 2600

#define FREQUENCY 50

```

```

void setup()
{
    pwm.begin();
    pwm.setPWMFreq(FREQUENCY);
}

int pulseWidth(int angle)
{
    int pulse_wide, analog_value;
    pulse_wide    =    map(angle,    0,    180,    MIN_PULSE_WIDTH,
MAX_PULSE_WIDTH);
    analog_value = int(float(pulse_wide) / 1000000 * FREQUENCY * 4096);
    return analog_value;
}

void loop() {
    pwm.setPWM(0, 0, pulseWidth(0));
    pwm.setPWM(1, 0, pulseWidth(0));
    pwm.setPWM(2, 0, pulseWidth(0));
    pwm.setPWM(3, 0, pulseWidth(0));
    pwm.setPWM(4, 0, pulseWidth(0));
    pwm.setPWM(5, 0, pulseWidth(0));
    delay(1000);

    pwm.setPWM(0, 0, pulseWidth(80));
    pwm.setPWM(1, 0, pulseWidth(80));
    pwm.setPWM(2, 0, pulseWidth(80));
    pwm.setPWM(3, 0, pulseWidth(80));
    pwm.setPWM(4, 0, pulseWidth(80));
    pwm.setPWM(5 , 0, pulseWidth(80));

```

```

delay(1000);
}

```

Після остаточної сборки механічної частини моделі та апаратної частини написано програму для обробки сигналів та передачі їх на механічну частину. Головний пристрій отримує сигнали з датчиків, передані через модуль бездротової передачі даних та перевіряє їх. Пристрій перевіряє розташування всіх сервоприводів, обробляє отриманий сигнал та перетворює їх у нові кути для зміни положення сервоприводів. Лістинг програми наведено у додатку Д.

Система доволі швидко обробляє дані і змінює положення моделі, а більш точного розпізнавання рухів можна домогтись, використовуючи більшу кількість датчиків ЕМГ.

					ДП 6416.03.000 ПЗ	Арк.
						58
Змн.	Арк.	№ докум.	Підпис	Дата		

## ВИСНОВКИ ДО РОЗДІЛУ 4

На даному етапі розроблено модель роботизованої руки, що отримує сигнали від ЕМГ датчиків, які розташовані на реальній руці. Модель обробляє отримані сигнали та повторює рухи руки.

Було зібрано механічну частину моделі та реалізовано систему передачі рухів від сервоприводів. Протестовано роботу ЕМГ датчика та сервоприводів разом з ШІМ контролером.

Розроблено схеми пристроїв та зібрана апаратна частина. Остаточного розроблено кінцевий алгоритм та написано програму за ним. Система працює доволі швидко, а кращого розпізнавання рухів можна домогтись використовуючи більшу кількість ЕМГ датчиків.

					ДП 6416.03.000 ПЗ	Арк.
						59
Змн.	Арк.	№ докум.	Підпис	Дата		

## ЛІТЕРАТУРА

1. Advance of the robots [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/by6T2al>.
2. Nourbakhsh I. Robot Futures / Illah Reza Nourbakhsh.
3. Робототехніка [Електронний ресурс] – Режим доступу до ресурсу: <https://iot.ru/wiki/robototekhnika>
4. Роботи [Електронний ресурс] – Режим доступу до ресурсу: <https://rb.ru/story/robotics-dictionary/>.
5. Управління роботами [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/yu6Tt4V>.
6. Інтерфейси управління [Електронний ресурс] – Режим доступу до ресурсу: <http://robotrends.ru/robopedia/interfyaasy-dlya-upravleniya-robotami>.
7. ЕМГ [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/5y6TUj4>.
8. ЕМГ [Електронний ресурс] – Режим доступу до ресурсу: <https://studfile.net/preview/3800608/>.
9. Shadow hand [Електронний ресурс] – Режим доступу до ресурсу: <https://www.shadowrobot.com/products/dexterous-hand/>.
10. I-Limb [Електронний ресурс] – Режим доступу до ресурсу: <https://novate.ru/blogs/220413/22900/>
11. BeBionic [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/ry6YmBl>
12. ДСТУ 2938-94. Системи оброблення інформації. Основні поняття. Терміни та визначення.
13. ДСТУ 3956-2000. Технічні засоби вимірювання та керування у промислових процесах. Частина 1. Основні поняття. Терміни та визначення.
14. N. E. Huang, Z. Shen, S. R. Long, M. L. Wu, H. H. Shih, Q. Zheng, N. C. Yen, C. C. Tung, and H. H. Liu, "The empirical mode decomposition and Hilbert spectrum for nonlinear and non-stationary time series analysis", vol. 454, pp. 903–995, 1998.

					ДП 6416.03.000 ПЗ	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дата		

15. Hilbert-Huang Transform [Електронний ресурс] – Режим доступу до ресурсу: <https://www.sciencedirect.com/topics/engineering/hilbert-huang-transform>.
16. G. Rilling, P. Flandrin, and P. Gonçalves, "On empirical mode decomposition and its algorithms", Grado, Italy, June 2003.
17. N. E. Huang and S. S. P. Shen, Hilbert-Huang Transform and its Applications, World Scientific Publishing Co. Pte Ltd, 2005.
18. D. Joshi, K. Kandpal, and S. Anand. "Feature evaluation to reduce false triggering in threshold based emg prosthetic hand." International Conference on Biomedical Engineering, pp. 769-772, 2008.
19. Kurtosis [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/Ny6R6VS>.
20. A. Phinyomark, P. Phukpattaranont, C. Limsakul, "Feature reduction and selection for EMG signal classification", Expert Systems with Applications 39, Elsevier, 2012.
21. S. Theodoridis, K. Koutroumbas, "Pattern Recognition", Elsevier, 4th ed., 2009.
22. K. Kira and L.A. Rendell, "The Feature Selection Problem: Traditional Methods and a New Algorithm", Proc. 10th Nat'l Conf. Artificial Intelligence, pp. 129-134, 1992.
23. L. Smith, "A tutorial on Principal Components Analysis", 2002.
24. Principal Component Analysis [Електронний ресурс] – Режим доступу до ресурсу: [http://www.nlpca.org/pca\\_principal\\_component\\_analysis.html](http://www.nlpca.org/pca_principal_component_analysis.html).
25. I. Kononenko, "Estimating attributes: analysis and extensions of RELIEF". In Machine Learning: ECML-94 pp. 171-182, 1994.
26. ARDUINO UNO REV3 [Електронний ресурс] – Режим доступу до ресурсу: <http://arduino.cc/en/Main/arduinoBoardUno>.
27. SPI library [Електронний ресурс] – Режим доступу до ресурсу: <https://www.arduino.cc/en/reference/SPI>.
28. How Do Servo Motors Work [Електронний ресурс] – Режим доступу до ресурсу: <https://cutt.ly/Gy6RnpL>.
29. MyoWare™ Muscle Sensor (AT-04-001) DATASHEET [Електронний ресурс] – Режим доступу до ресурсу: <https://cdn.sparkfun.com/assets/a/3/a/f/a/AT-04-001.pdf>.

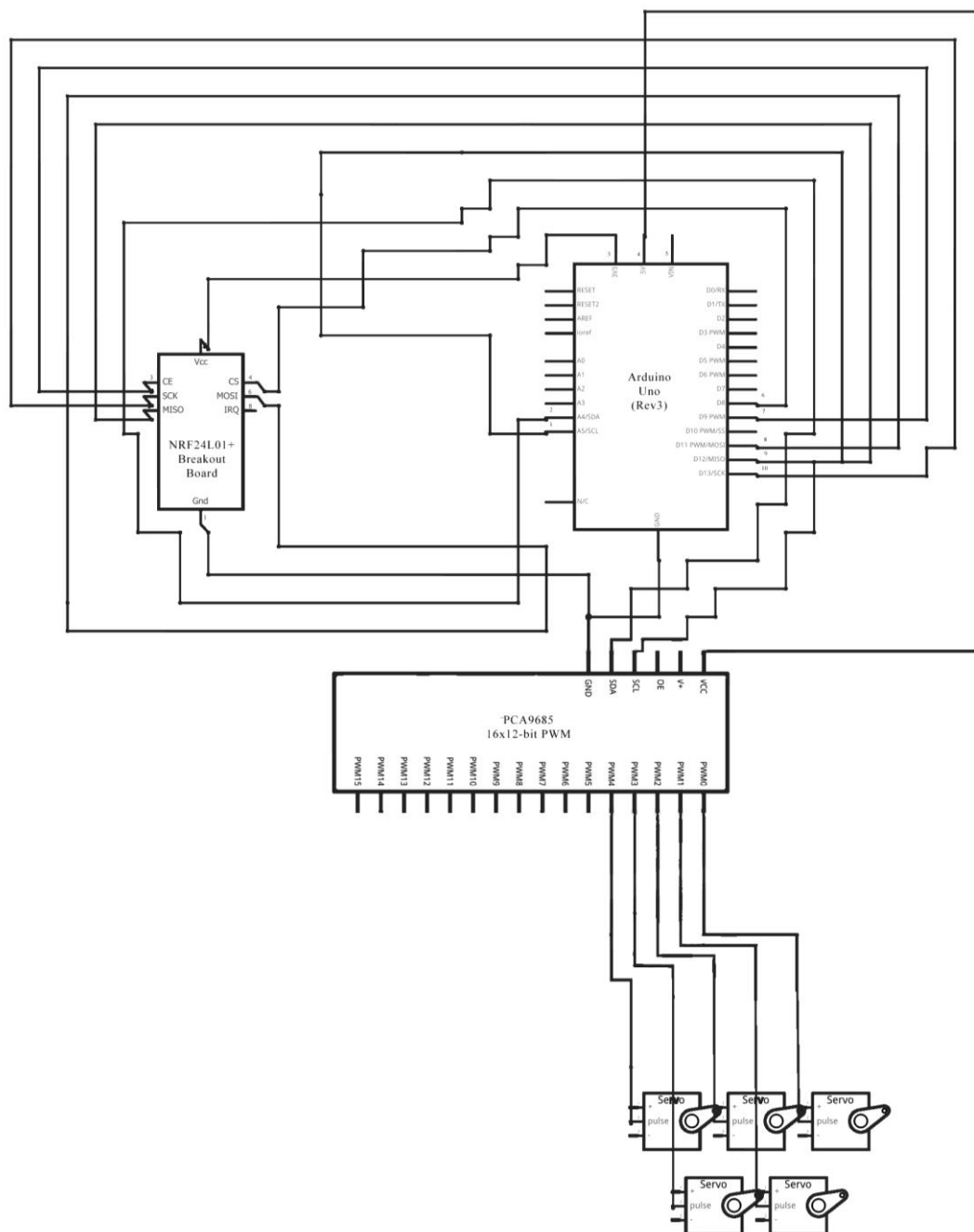
## ДОДАТОК А

Роботизована рука

Схема пристрою 1

Аркушів 1

Київ – 2020 року



					ДП 6416.04.000 Д1			
Зм.	Арк.	№ докум.	П	Да	<p>Роботизована рука</p> <p>Схема пристрою 1</p>			
Розроб.	Невмержицька А.Ю.							
Перевір.	Габінет А.В.							
Н. Контр.	Симоненко В.П.							
Затверд.	Габінет А.В.				<p>Літ.</p> <p>Арк.</p> <p>Аркушів</p> <p>1</p> <p>1</p> <p>НТУУ "КПІ", ФІОТ, каф. ОТ, гр. ІО-64</p>			



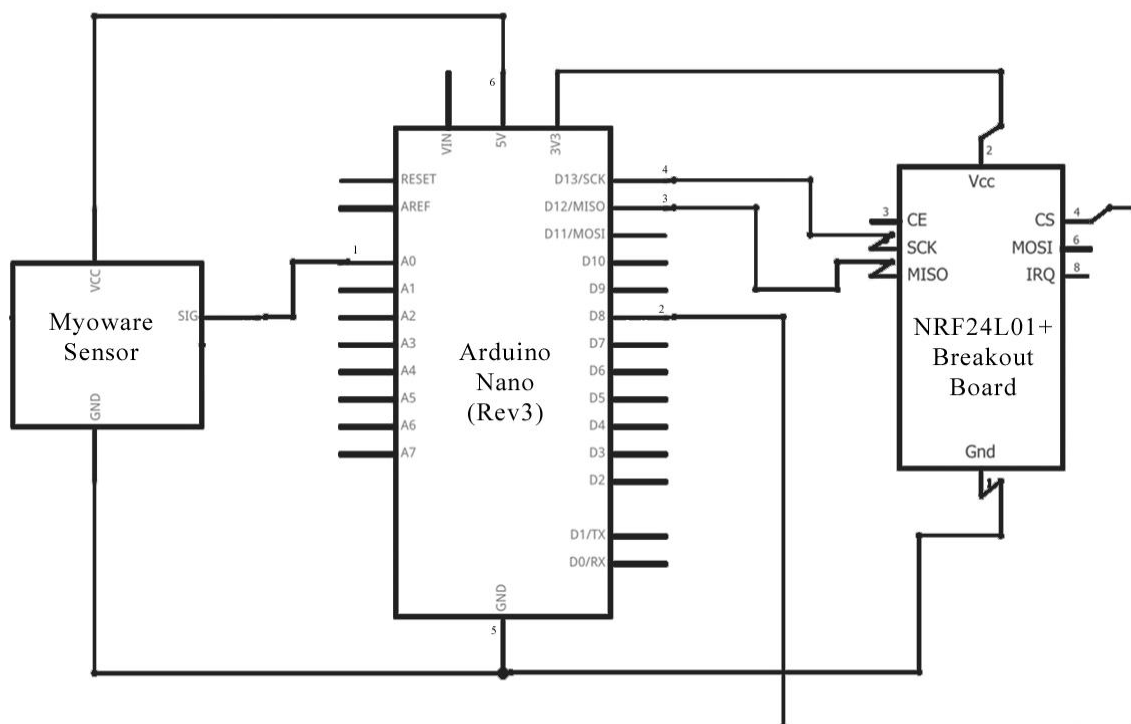
## ДОДАТОК Б

Роботизована рука

Схема пристрою 2

Аркушів 1

Київ – 2020 року



					ДП 6416.05.000 Д2						
Зм.	Арк.	№ докум.	П	Да							
Розроб.		Невмержицька А.Ю.			Роботизована рука Схема пристрою 2			Літ.	Арк.	Аркушів	
Перевір.		Габінет А.В.								1	1
								НТУУ “КПІ”, ФІОТ, каф. ОТ, гр. ІО-64			
Н. Контр.		Симоненко В.П.									
Затверд.		Габінет А.В.									

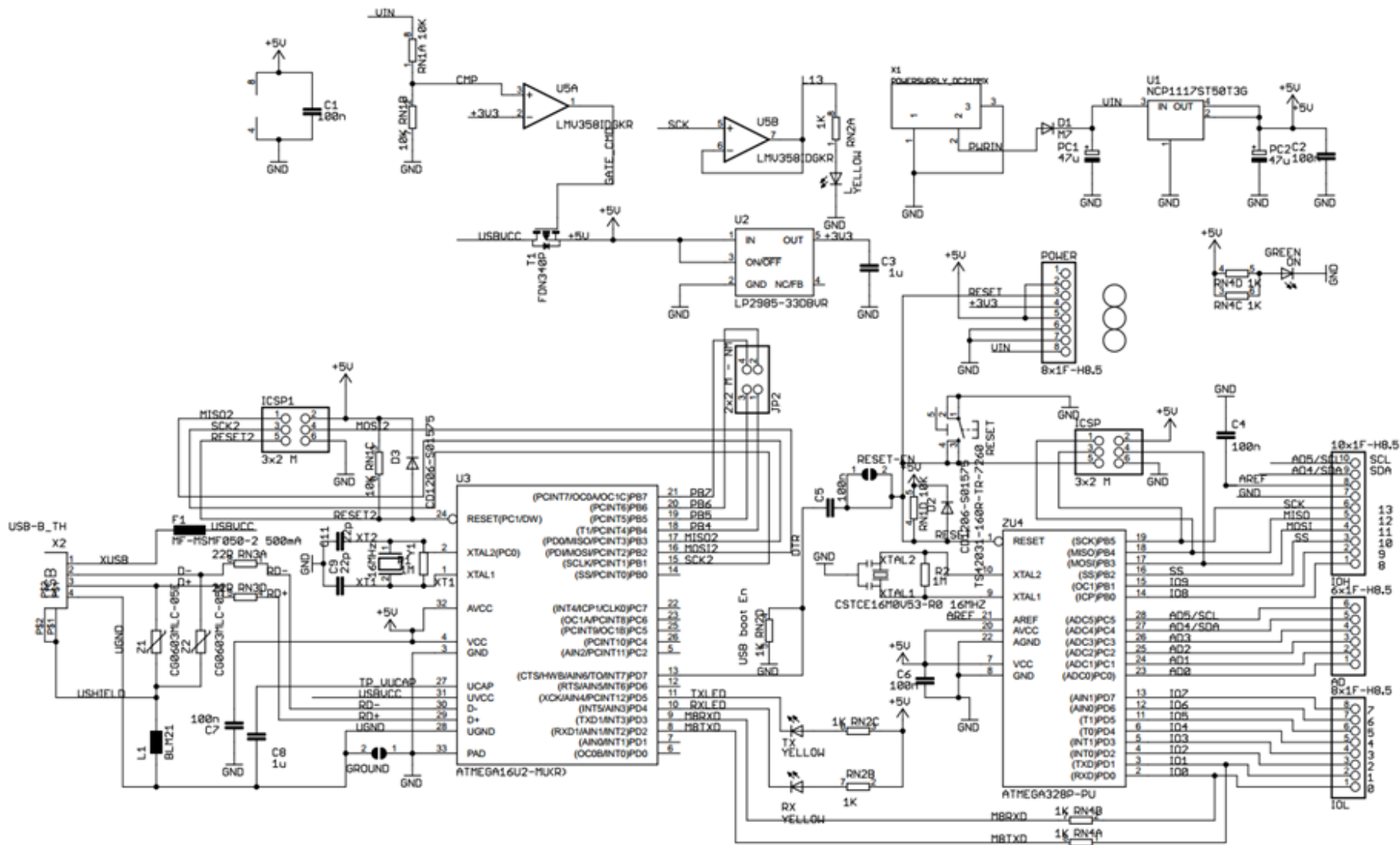
## ДОДАТОК В

Роботизована рука

Схема Arduino Uno

Аркушів 1

Київ – 2020 року



Зм	Арк	№ докум	Підпис	Дата
Розроб	Немержицька А.Ю.			
Перевірів	Габінет А.В.			
Н. Контр	Симоненко В.П.			
Затверд.				

ДП 6416.05.000 Д2

Роботизована рука  
Структурна схема  
Arduino Uno

Літ	Аркуш	Аркушів
	1	1

НТУУ «КПІ» ФІОТ  
ГРУПА ІО-64

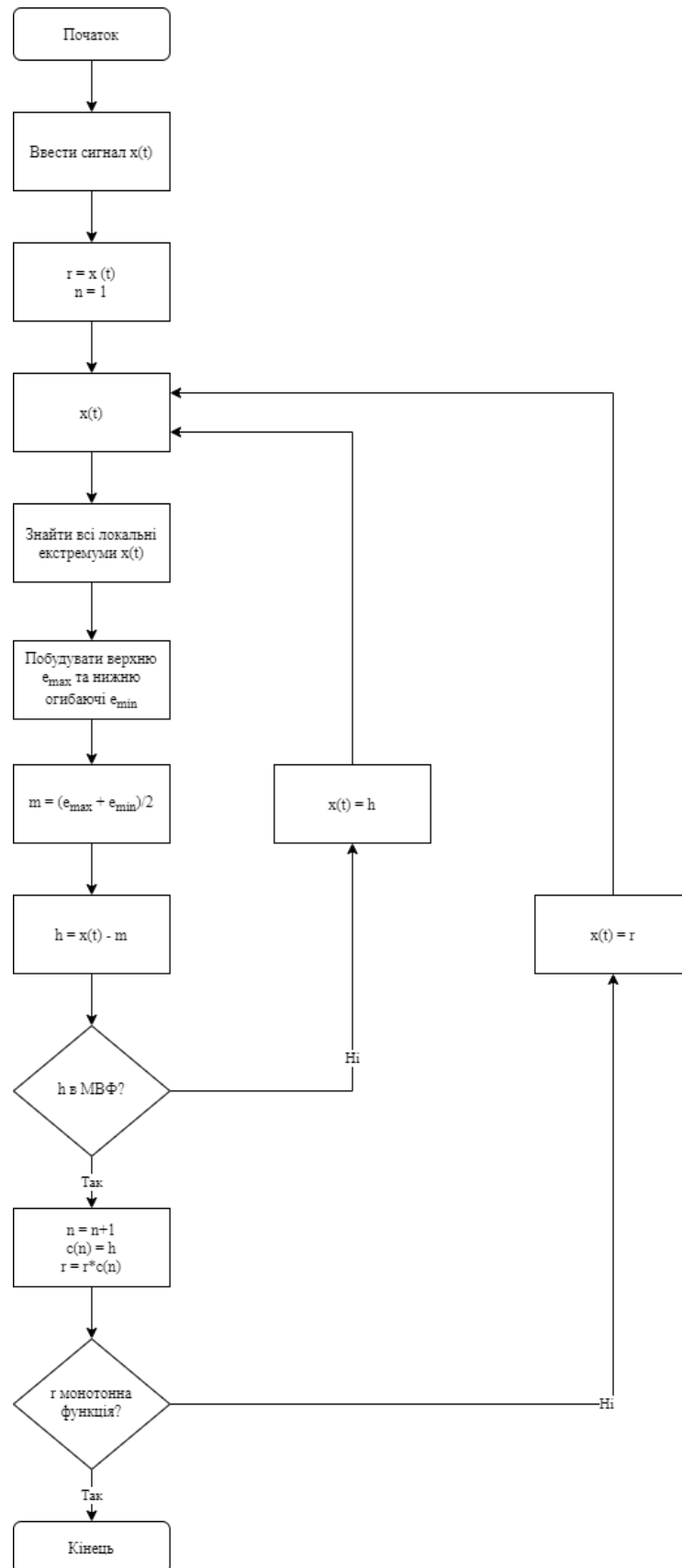
## ДОДАТОК Г

Роботизована рука

Алгоритм ЕМД

Аркушів 1

Київ – 2020 року



					ДП 6416.07.000 Д4								
Зм.	Арк.	№ докум.	П	Да									
Розроб.		Невмержицька А.Ю.			Роботизована рука Алгоритм ЕМД				Літ.	Арк.	Аркушів		
Перевір.		Габінет А.В.									1	1	
Н. Контр.		Симоненко В.П.							НТУУ “КПІ”, ФІОТ, каф. ОТ, гр. ІО-64				
Затверд.		Габінет А.В.											

## ДОДАТОК Д

Роботизована рука

Лістинг програми

Аркушів 4

Київ – 2020 року

```

#include <SPI.h>
#include <RF24.h>
#include <RF24_config.h>
#include <nRF24L01.h>

#include
<Adafruit_PWMServoDriver.h>
#include <Wire.h>

#define SERVOMIN 150
#define SERVOMAX 600

Adafruit_PWMServoDriver pwm =
Adafruit_PWMServoDriver();

int rf_cen = 9; pin

int rf_cs = 8;

RF24 rf(rf_cen, rf_cs);

uint8_t pipe_rx[8] = {0x0E, 0xE6,
0x0D, 0xA7, 0, 0, 0, 0};

uint8_t swapbits(uint8_t a){
    uint8_t v = 0;
    if(a & 0x80) v |= 0x01;
    if(a & 0x40) v |= 0x02;
    if(a & 0x20) v |= 0x04;
    if(a & 0x10) v |= 0x08;
    if(a & 0x08) v |= 0x10;
    if(a & 0x04) v |= 0x20;
    if(a & 0x02) v |= 0x40;
    if(a & 0x01) v |= 0x80;
    return v;
}

long last_servo_upd = 0;
byte in_pack[32];

```

					ДП 6416.08.000 Д5								
Зм.	Арк.	№ докум.	П	Да									
Розроб.		Невмержицька А.Ю.			Роботизована рука Лістинг програми				Літ.	Арк.	Аркушів		
Перевір.		Габінет А.В.									1	4	
Н. Контр.		Симоненко В.П.											
Затверд.		Габінет А.В.			НТУУ “КПІ”, ФІОТ, каф. ОТ, гр. ІО-64								



```

unsigned long unit_ids[3] =
{4294963881, 4294943100, 28358};

int unit_vals[3] = {0, 0, 0};

float tgt_angles[5];
float cur_angles[5];

float angle_open = 30;
float angle_closed = 150;

void setup() {
    SPI.begin();
    SPI.setBitOrder(MSBFIRST);

    SPI.beginTransaction(SPISettings(1
000000, MSBFIRST,
SPI_MODE0));

    for(int x = 0; x < 8; x++)
        pipe_rx[x] =
swapbits(pipe_rx[x]);
    rf.begin();

    rf.setDataRate(RF24_1MBPS);
    rf.setAddressWidth(4);
    rf.setChannel(22);
    rf.setRetries(0, 0);
    rf.setAutoAck(0);
    rf.disableDynamicPayloads();
    rf.setPayloadSize(32);
    rf.openReadingPipe(0, pipe_rx);

    rf.setCRCLength(RF24_CRC_DISABLED);
    rf.disableCRC();
    rf.startListening();

    Serial.begin(115200);
    pwm.begin();
    pwm.setPWMFreq(60);
    for(int i = 0; i < 5; i++)
    {
        tgt_angles[i] = angle_open;
        cur_angles[i] = angle_open;
    }
}

```

```

void setAngle(int n, float angle){
    pwm.setPWM(n, 0, SERVOMIN +
angle * 0.005556 * (SERVOMAX -
SERVOMIN));
}

float angle_speed = 15
float v0 = 0, v1 = 0, v2 = 0;

void loop()
{
    if(rf.available())
    {
        rf.read(in_pack, 32);
        byte u1 = in_pack[3];
        byte u2 = in_pack[4];
        byte u3 = in_pack[5];
        byte u4 = in_pack[6];
        unsigned long id = (u1<<24) |
(u2<<16) | (u3<<8) | u4;
        if(in_pack[7] != 32) id = 0;
        int val = in_pack[10];
        if(val != in_pack[11]) id = 0;
        for(int n = 0; n < 3; n++)
            if(id == unit_ids[n])
                unit_vals[n] = val;
    }
    long ms = millis();
    if(ms - last_servo_upd > 20)
    {
        last_servo_upd = ms;
        for(int n = 0; n < 5; n++)
        {
            if(cur_angles[n] < tgt_angles[n]
- angle_speed/2) cur_angles[n] +=
angle_speed;
            if(cur_angles[n] > tgt_angles[n]
+ angle_speed/2) cur_angles[n] -=
angle_speed;
        }
        for(int n = 0; n < 5; n++)
            setAngle(n, cur_angles[n]);

        v0 = v0*0.7 +
0.3*(float)unit_vals[0];
        v1 = v1*0.7 +
0.3*(float)unit_vals[1];
    }
}

```

```

v2 = v2*0.7 +
0.3*(float)unit_vals[2];

float scor0 = 4.0*v0*v0/((v1*0.3
+ 20)*(v2*1.3 + 15));

float scor1 = 4.0*v1*v1/((v0*2.0
+ 20)*(v2*2.0 + 20));

float scor2 = 4.0*v2*v2/((v0*1.2
+ 20)*(v1*0.5 + 15));

Serial.print(scor0);
Serial.print(' ');
Serial.print(scor1);
Serial.print(' ');
Serial.println(scor2);

if(scor2 < 0.5)
    tgt_angles[0] = angle_open;
if(scor2 > 1.0)
    tgt_angles[0] = angle_closed;

if(scor1 < 0.5)

```

```

{
    tgt_angles[1] = angle_open;
    tgt_angles[2] = angle_open;
}
if(scor1 > 1.0)
{
    tgt_angles[1] = angle_closed;
    tgt_angles[2] = angle_closed;
}

if(scor0 < 0.5)
{
    tgt_angles[3] = angle_open;
    tgt_angles[4] = angle_open;
}
if(scor0 > 1.0)
{
    tgt_angles[3] = angle_closed;
    tgt_angles[4] = angle_closed;
}
}
}

```

					ДП 6416.08.000 Д5	Арк.
						4
Змн.	Арк.	№ докум.	Підпис	Дата		